IQ80960RM/RN Evaluation Platform

Board Manual

February 1999

Order Number: 273160-004

Contents

Figures

Tables

Introduction 1

This manual describes the IQ80960RM and IQ80960RN evaluation platforms for Intel's i960® RM/RN I/O processor. The i960 RM/RN I/O processors combine an 80960JT core with two PCI bus interfaces, as well as a memory controller, DMA channels, an interrupt controller interface, and an I^2C serial bus. The difference between the two processors is that the 80960RN utilizes 64-bit primary PCI and secondary PCI buses while the 80960RM utilizes both a 32-bit primary and secondary PCI bus. The IQ80960RM and IQ80960RN platforms are full-length PCI adapter boards and are 8.9" in height to accommodate four standard PCI connectors on the secondary PCI bus. The boards can be installed in any PCI host system that complies with the *PCI Local Bus Specification* Revision 2.1. PCI devices can be connected to the secondary bus to build powerful intelligent I/O subsystems.

Figure 1-1. IQ80960RM/IQ80960RN Platform Functional Block Diagram

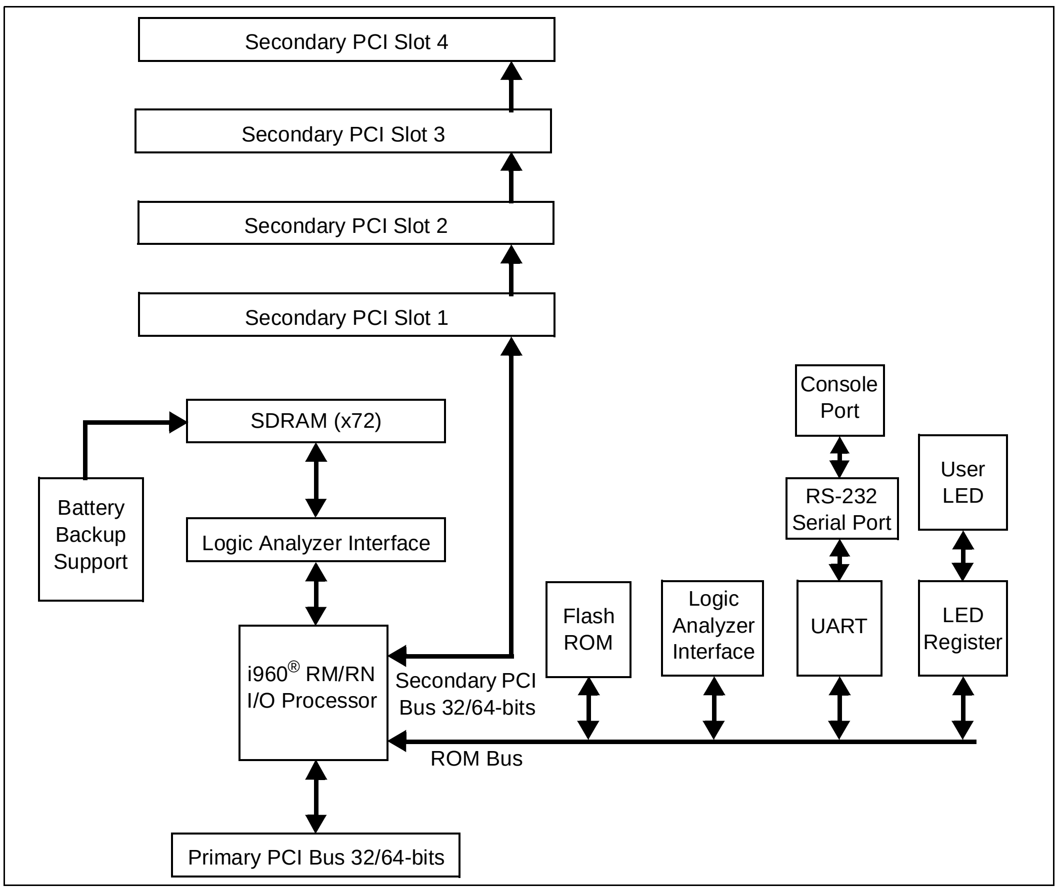
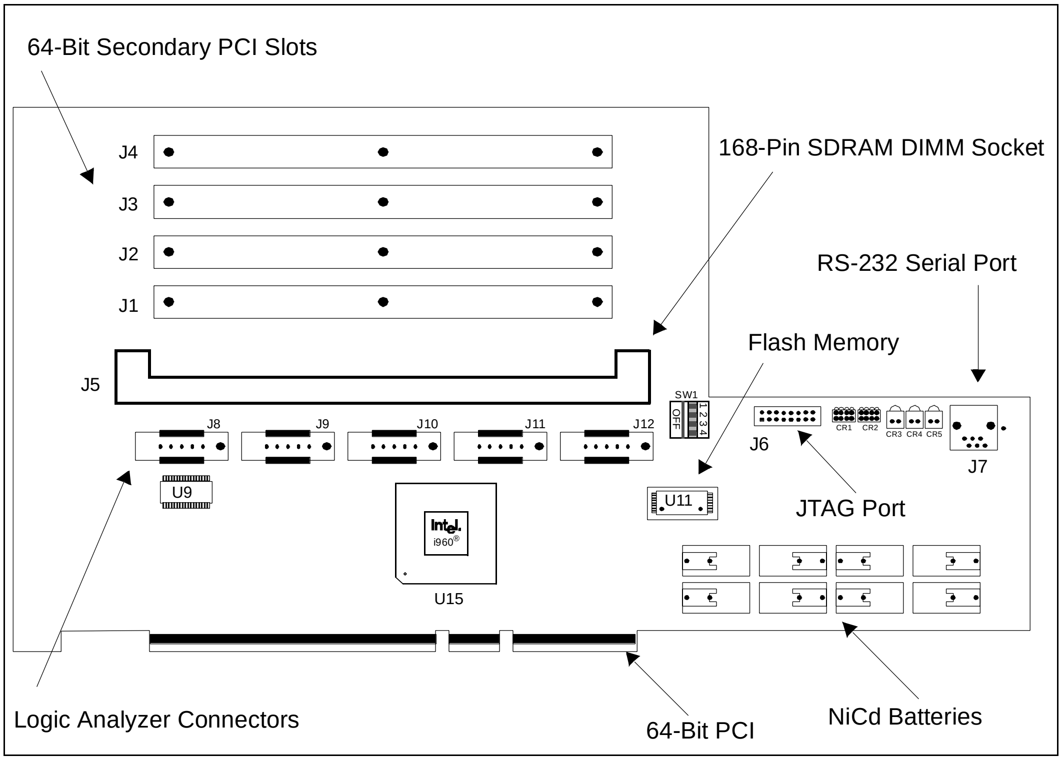


Figure 1-2. IQ80960RN Platform Physical Diagram



1.1 i960® RM/RN I/O Processor and IQ80960RM/RN Features

The i960 RM/RN I/O processor serves as the main component of a high performance, PCI-based intelligent I/O subsystem. The IQ80960RM and IQ80960RN platforms allow the developer to connect PCI devices to the i960 RM/RN I/O processors using the four secondary PCI expansion connectors. The features of the IQ80960RM and IQ80960RN platforms are enumerated below and shown in Figure 1-1 and Figure 1-2.

- i960 RM/RN I/O processor
- Modified PCI long-card form factor
- 64-bit or 32-bit primary PCI bus interface (80960RM 32-bit only)
- 64-bit or 32-bit secondary PCI bus connected to the primary PCI interface with a PCI-to-PCI bridge (80960RM 32-bit only)
- DMA channels on both PCI buses
- I^2C Serial Bus
- 168-pin, 3.3V DIMM socket supporting 16 to 128 Mbytes of Synchronous DRAM organized x72 to support Error Correction Code (ECC) and clocked at 66 MHz (ships with 16 M/ECC installed)
- Serial console port based on 16C550 UART
- Eight user-programmable LEDs
- 3 Indicator LEDs: processor has passed self-test, 3.3 V is supplied to SDRAM, and 3.3 V is supplied to secondary PCI slots
- Flash ROM, 2 Mbytes
- Logic analyzer connectors for SDRAM bus, ROM bus and secondary PCI arbitration signals
- Fan heatsink monitor circuit
- Battery backup for SDRAM
- JTAG header

1.2 Software Development Tools

A number of software development tools are available for the i960® processor family[1]. This manual provides information on two software development toolsets: Wind River System's Tornado* for I_20* and Intel's CTOOLS. If you are using other software development tools, read through the information in this chapter and in Chapter 2 to gain a general understanding of how to use your tools with this board.

1. To view the electronic tools catalog, access http://developer.intel.com/design/develop.htm/ from the web.

1.3 Tornado* for I20* Software Development Toolset

Tornado for $I_2$0 is a complete toolset featuring an integrated development environment including a compiler, assembler, linker, and debugger. It also features a real-time operating system.

1.3.1 IxWorks* Real-Time Operating System

The IQ80960RM/RN platforms are equipped with Wind River Systems, Inc.'s IxWorks*. IxWorks provides for the elements of the I_2O standard: an event-driven driver framework, host message protocols, and executive modules for configuration and control. IxWorks also allows for the writing of basic device drivers and provides NOS-to-driver independence. TORNADO for I_2O provides a visual environment for building, testing and debugging of I_2O drivers.

1.3.2 TORNADO Build Tools

TORNADO for I_2O includes a collection of supporting tools that provide a complete development tool chain. These include the compiler, assembler, linker and binary utilities. Also provided is an I_2O module builder, which creates I_2O-loadable modules.

1.3.3 TORNADO Test and Debug Tools

TORNADO for I_2O test and debug tools include the dynamic loader, the CrossWind* debugger, the WindSh* interactive shell, and a system browser.

The dynamic loader allows for interactive loading, testing, and replacement of individual object modules that comprise a driver.

CrossWind is an extended version of GDB960. Using it you can debug I_2O drivers by setting breakpoints on desired I_2O components. A variety of windows display source code, registers, locals, stack frame, memory and so on.

WindSh allows you to communicate to the IQ80960RM/RN platform via an RS-232 serial port. The IQ80960RM/RN platform supports port speeds from 300 to 115,200 bps. The shell can be used to:

- control and monitor I_2O drivers
- format, send and receive driver messages
- examine hardware registers
- run automated I_2O test suites

The shell also provides essential debugging capabilities; including breakpoints, single stepping, stack checking, and disassembly.

1.4 CTOOLS Software Development Toolset

Intel's i960 processor software development toolset, CTOOLS, features advanced C/C++ - language compilers for the i960 processor family. CTOOLS development toolset is available for Windows* 95/NT-based systems and a variety of UNIX workstation hosts. These products provide execution profiling and instruction scheduling optimizations and include an assembler, a linker, and utilities designed for embedded processor software development.

1.4.1 CTOOLS and the MON960 Debug Monitor

In place of IxWorks, the IQ80960RM/RN platform can be equipped with Intel's MON960, an on-board software monitor that allows you to execute and debug programs written for i960 processors in a non-I^2O environment. The monitor provides program download, breakpoint, single step, memory display, and other useful functions for running and debugging a program.

The IQ80960RM/RN platform works with the source-level debuggers provided with CTOOLS, including GDB960 (command line version) and GDB960V (GUI version).

1.4.1.1 MON960 Host Communications

MON960 allows you to communicate and download programs developed for the IQ80960RM/RN platform across a host system's serial port or PCI interface. The IQ80960RM/RN platform supports two methods of communication: terminal emulation and Host Debugger Interface (HDI).

1.4.1.2 Terminal Emulation Method

Terminal emulation software on your host system can communicate to MON960 on the IQ80960RM/RN platform via an RS-232 serial port. The IQ80960RM/RN platform supports port speeds from 300 to 115,200 bps. Serial downloads to MON960 require that the terminal emulation software support the XMODEM protocol.

Configure the serial port on the host system for 300-115,200 baud, 8 bits, one stop bit, no parity with XON/XOFF flow control.

1.4.1.3 Host Debugger Interface (HDI) Method

You may use a source-level debugger, such as Intel's GDB960 and GDB960V to establish serial or PCI communications with the IQ80960RM/RN platform. The MON960 Host Debugger Interface (HDI) provides a defined messaging layer between MON960 and the debugger. For more information on this interface, see the *MON960 Debug Monitor User's Manual* (484290).

HDI connection requests cannot be detected by MON960 if the user has already initiated a connection using a terminal emulator. In this case, the IQ80960RM/RN platform must be reset before the debugger can connect to MON960.

1.5 SPI610 JTAG Emulation System

The SPI610 JTAG Emulation System from Spectrum Digital, Inc. is included in the IQ80960RM/RN development kit. It furnishes the default host development environment-to-evaluation board communication link based on the i960 RM/RN I/O processor JTAG interface.

Refer to the SPI610 Reference Manual for JTAG emulation system installation and operation for both the Tornado and CTOOLS environment. Optionally, evaluation board serial port communications can be used for this communication link (see Section 1.3.3, "TORNADO Test and Debug Tools" on page 1-4).

1.6 About This Manual

A brief description of the contents of this manual follows.

Chapter 1, "Introduction"	Introduces the IQ80960RM and IQ80960RN Evaluation Board features. This chapter also describes Intel's CTOOLS* and WindRiver Systems IxWorks* software development tools, and defines notational-conventions and related documentation.
Chapter 2, "Getting Started"	Provides step-by-step instructions for installing the IQ80960RM or IQ80960RN platform in a host system and downloading and executing an application program. This chapter also describes Intel's software development tools, the MON960 Debug Monitor, IxWORKS, software installation, and hardware configuration.
Chapter 3, "Hardware Reference"	Describes the locations of connectors, switches and LEDs on the IQ80960RM and IQ80960RN platforms. Header pinouts and register descriptions are also provided in this chapter.
Chapter 4, "i960® RM/RN I/O Processor Overview"	Presents an overview of the capabilities of the i960 RM/RN I/O processor and includes the CPU memory map.
Chapter 5, "MON960 Support for IQ80960RM/RN"	Describes a number of features added to MON960 to support application development on the i960 RM/RN I/O processor.
Appendix A, "Bill of Materials"	Shows complete parts list IQ80960RM and IQ80960RN Evaluation Platforms.
Appendix B, "Schematics"	Complete set of schematics for the IQ80960RM and IQ80960RN Evaluation Platforms.
Appendix C, "PLD Code"	Example PLD code used on IQ80960RM and IQ80960RN evaluation boards for SDRAM battery backup.
Appendix D, "Recycling the Battery"	Information on the RBRC program and the locations of participating recycling centers.

1.7 Notational-Conventions

The following notation conventions are consistent with other i960 RM/RN I/O processor documentation and general industry standards.

# or overbar	In code examples the pound symbol (#) is appended to a signal name to indicate that the signal is active. Normally inverted clock signals are indicated with an overbar above the signal name (e.g., RAS).
Bold	Indicates user entry and/or commands. PLD signal names are in bold lowercase letters (e.g., **h_off**, **h_on**).
Italics	Indicates a reference to related documents; also used to show emphasis.
`Courier` font	Indicates `code` examples and file directories and names.
Asterisks (*)	On non-Intel company and product names, a trailing asterisk indicates the item is a trademark or registered trademark. Such brands and names are the property of their respective owners.
UPPERCASE	In text, signal names are shown in uppercase. When several signals share a common name, each signal is represented by the signal name followed by a number; the group is represented by the signal name followed by a variable *(n)*. In code examples, signal names are shown in the case required by the software development tool in use.
Designations for hexadecimal and binary numbers	In text, instead of using subscripted "base" designators (e.g., FF_{16}) or leading "0x" (e.g., 0xFF) hexadecimal numbers are represented by a string of hex digits followed by the letter *H*. A zero prefix is added to numbers that begin with *A* through *F.* (e.g., *FF* is shown as *0FFH*.) In examples of actual code, "0x" is used. Decimal and binary numbers are represented by their customary notations. (e.g., 255 is a decimal number and 1111 1111 is a binary number. In some cases, the letter *B* is added to binary numbers for clarity.)

1.8 Technical Support

Up-to-date product and technical information is available electronically from:

- Intel's World-Wide Web (WWW) Location: http://www.intel.com
- IQ80960RM and IQ80960RN Product Information: http://developer.intel.com/design/i960

For technical assistance, electronic mail (e-mail) provides the fastest route to reach engineers specializing in IQ80960RM and IQ80960RN issues. Posting messages on the Embedded Microprocessor Forum at http://support.intle.com/newsgroups/ is also a direct route for IQ80960RM and IQ80960RN technical assistance. See Section 1.8.2.

Within the United States and Canada you may contact the Intel Technical Support Hotline. See Section 1.8.1 for a list of customer support sources for the US and other geographical areas.

1.8.1 Intel Customer Electronic Mail Support

For direct support from engineers specialing in i960® Microprocessor issues send e-mail in english to 960tools@intel.com.

Questions and other messages may be posted to the Embedded Microprocessor Forum at http://support.intel.com/newsgroups/.

1.8.2 Intel Customer Support Contacts

Contact Intel Corporation for technical assistance for the IQ80960RM/RN evaluation platform.

Country	Literature	Customer Support Number
United States	800-548-4725	800-628-8686
Canada	800-468-8118 or 303-297-7763	800-628-8686
Europe	Contact local distributor	Contact local distributor
Australia	Contact local distributor	Contact local distributor
Israel	Contact local distributor	Contact local distributor
Japan	Contact local distributor	Contact local distributor

1.8.3 Related Information

To order printed manuals from Intel, contact your local sales representative or Intel Literature Sales (1-800-548-4725).

Table 1-1. Document Information

Product	Document Name	Company/ Order #
All	*Developers' Insight CD-ROM*	Intel # 273000
80960RM/RN	*i960® RM/RN I/O Processor Developer's Manual*	Intel # 273158
	80960RM I/O Processor Data Sheet	Intel # 273156
	80960RN I/O Processor Data Sheet	Intel # 273157
	i960® RM/RN I/O Processor Design Guide	Intel # 273139
	MON960 Debug Monitor User's Guide	Intel #484290
	PCI Local Bus Specification Revision 2.1	PCI Special Interest Group 1-800-433-5177
	Writing I_2O Device Drivers in IxWorks	Wind River Systems, Inc. #DOC-1173-8D-02
	IxWorks Reference Manual	Wind River Systems, Inc. #DOC-1173-8D-03
	VxWorks Programmer's Guide	Wind River Systems, Inc. #DOC-11045-ZD-01
	Tornado User's Guide	Wind River Systems, Inc. #DOC-1116-8D-01
	Tornado for I_2O	Wind River Systems, Inc. #DOC-12381-8D-00
	Tornado for I_2O Compact Disk Rev. 1.0	#TDK-12380-ZC-00
	SP610 Emulation System Reference Manual	Spectrum Digital Inc. # 503715

Contact Cyclone Microsystems for additional information about their products and literature:

Table 1-2. Cyclone Contacts

Cyclone Microsystems **25 Science Park** **New Haven CT 06511**	Phone: 203-786-5536
	FAX: 203-786-5025
	e-mail: info@cyclone.com WWW: http://www.cyclone.com

Getting Started 2

This chapter contains instructions for installing the IQ80960RM/RN platform in a host system and, how to download and execute an application program using Wind River System's IxWorks∗ or Intel's CTOOLS software development toolsets.

2.1 Pre-Installation Considerations

This section provides a general overview of the components required to develop and execute a program on the IQ80960RM/RN platform. IQ80960RM/RN evaluation boards support two software development toolsets, Wind River System's IxWorks and Intel's CTOOLS.

IxWorks is a complete toolset featuring an integrated development environment including a compiler, assembler, linker, and debugger. It also features a real-time operating system. If you are using the IxWorks operating system with the TORNADO* development environment, refer to the Wind River Systems, Inc. documentation referenced in Section 1.8.3.

CTOOLS is a complete C/C++-language software-development toolset for developing embedded applications to run on i960 processors. It contains a C/C++ compiler, the gcc960 and ic960 compiler driver programs, an assembler, runtime libraries, a collection of software-development tools and utilities, and printed and on-line documentation. The *MON960 Debug Monitor User's Guide* fully describes the components of MON960, including MON960 commands, the Host Debugger Interface Library (HDIL), and the MONDB.EXE utility. If you are using MON960 and the CTOOLS toolset, refer to section Section 2.2.1, "Installing Software Development Tools" on page 2-1.

See Chapter 1 for more information on the IxWorks and CTOOLS features.

The IQ80960RM/RN evaluation boards are supplied with IxWorks intelligent real-time operating system pre-loaded into the on-board Flash. You also have the option of installing the MON960 debug monitor, which is required if you are using the CTOOLS debugging tools, GDB960, GDB960V, or MONDB. Section 3.3.1 describes the Flash ROM programming utility, which allows you to load MON960 onto the platform or re-load IxWorks.

2.2 Software Installation

2.2.1 Installing Software Development Tools

If you haven't done so already, install your development software as described in its manuals. All references in this manual to CTOOLS or CrossWind assume that the default directories were selected during installation. If this is not the case, substitute the appropriate path for the default path wherever file locations are referenced in this manual.

2.3 Hardware Installation

Follow these instructions to get your new IQ80960RM/RN platform running. Be sure all items on the checklist were provided with your IQ80960RM/RN.

Warning: Static charges can severely damage the IQ80960RM/RN platforms. Be sure you are properly grounded before removing the IQ80960RM/RN platform from the anti-static bag.

2.3.1 Battery Backup

Battery backup is provided to save any information in SDRAM during a power failure. The IQ80960RM/RN platform contains four AA NiCd batteries, a charging circuit and a regulator circuit. The batteries installed in the IQ80960RM/RN platform are rated at 600 mA/Hr.

SDRAM technology provides a simple way of enabling data preservation through the self-refresh command. When the processor receives an active Primary PCI reset it issues the self-refresh command and drives the SCKE signals low. Upon seeing this condition, a PAL on the IQ80960RM/RN platform holds SCKE low before the processor loses power. The batteries maintain power to the SDRAM and the PAL to ensure self-refresh mode. When the PAL detects PRST# returning to inactive state, the PAL releases the hold on SCKE.

The battery circuit can be disabled by removing the batteries. LED CR4 indicates when the SDRAMs have sufficient power. If the batteries remain in the evaluation platform when it is depowered and/or removed from the chassis, the batteries will maintain the SDRAM for approximately 30 hours. Once power is again applied, the batteries will be fully charged in about 4 hours.

2.3.2 Installing the IQ80960RM/RN Platforms in the Host System

If you are installing the IQ80960RM/RN platform for the first time, visually inspect the board for any damage that may have occurred during shipment. If there are visible defects, return the board for replacement. Follow the host system manufacturer's instructions for installing a PCI adapter. The IQ80960RM/RN platform is a full-length PCI adapter and requires a PCI slot that is free from obstructions. The IQ80960RM/RN platform is taller than specified in the *PCI Local Bus Specification* Revision 2.1. The extended height of the board will require you to keep the cover off of your PC. Refer to Chapter 3 for physical dimensions of the board.

2.3.3 Verify IQ80960RM/RN Platform is Functional

These instructions assume that you have already installed the IQ80960RM/RN platform in the host system as described in Section 2.3.2.

1. To connect the serial port for communicating with and downloading to the IQ80960RM/RN platform, connect the RS-232 cable (provided with the IQ80960RM/RN) from a free serial port on the host system to the phone jack-style connector on the IQ80960RM/RN platform.
2. Upon power-up, the red FAIL LED turns off, indicating that the processor has passed its self-test.
3. If you have IxWorks installed in the flash ROM, the user LEDs display the binary pattern 99H. In the IxWorks development environment, raw serial input/output is not used. Instead, the Wind DeBug (WDB) protocol is run over the serial port, to allow communication with Tornado development tools. If the terminal emulation package is running at 115,200 baud, the letters "WDB_READY" display prior to launching in the WDB serial protocol.

4. If you have MON960 installed in the flash ROM, press **<ENTER>** on a terminal connected to the IQ80960RM/RN platform to bring up the MON960 prompt. MON960 automatically adjusts its baud rate to match that of the terminal at start-up. At baud rates other than 9600, it may be necessary to press **<ENTER>** several times.

2.4 Creating and Downloading Executable Files

To download code to the IQ80960RM/RN platform running IxWorks, consult Wind River documentation on the supplied TORNADO for I_2O CD-ROM. To download code to the IQ80960RM/RN platform, your compiler produces an ELF-format object file.

To download code to the IQ80960RM/RN platform running CTOOLS, consult the CTOOLS documentation for information regarding compiling, linking, and downloading applications. During a download, MON960 checks the link address stored in the ELF file, and stores the file at that location on the IQ80960RM/RN platform. If the executable file is linked to an invalid address on the IQ80960RM/RN platform, MON960 aborts the download.

2.4.1 Sample Download and Execution Using GDB960

This example shows you how to use GBD960 to download and execute a file named `myapp` via the serial port.

- Invoke GDB960. From a Windows 95/NT command prompt, issue the command:

  ```
  gdb960 -r com2 myapp
  ```

  This command establishes communication and downloads the file `myapp`.
- To execute the program, enter the command from the GDB960 command prompt:

  ```
  (gdb960) run
  ```

More information on the GDB960 commands mentioned in this section can be found in the *GDB960 User's Manual*.

Hardware Reference 3

3.1 Power Requirements

The IQ80960RM/RN platform draws power from the PCI bus. The power requirements of the IQ80960RM/RN platforms are shown in Table 3-1 and Table 3-2. The numbers do not include the power required by a PCI card(s) mounted on one or more of the IQ80960RM/RN platforms' four expansion slots.

Table 3-1. IQ80960RN Platform Power Requirements

Voltage	Typical Current	Maximum Current
+3.3 V	0 V*	0 V*
+5 V	1.45 A	1.96 A
+12 V	286 mA	485 mA
-12 V	1 mA	1 mA

NOTE: Does not include the power required by a PCI card(s) mounted on the IQ80960RN platform.
* +3.3V for 80960RN Processor created on board from +5V.

Table 3-2. IQ80960RM Platform Power Requirements

Voltage	Typical Current	Maximum Current
+3.3 V	0 V*	0 V*
+5 V	1.32 A	1.86 A
+12 V	284 mA	485 mA
-12 V	1 mA	1 mA

NOTE: Does not include the power required by a PCI card(s) mounted on the IQ80960RM platform.
* +3.3V for 80960RM Processor created on board from +5V.

3.2 SDRAM

The IQ80960RM/RN platform is equipped with a 168-pin DIMM socket formatted to accept +3.3V synchronous DRAM with or without Error Correction Code (ECC). The socket will accept SDRAM from 8 Mbytes to 128 Mbytes. 128 Mbyte SDRAMs are available in both x64 and x72 configurations. Note that 8 Mbyte SDRAMs are only for x64 or non-ECC memory. The SDRAM is accessible from either of the PCI buses, via the ATUs, and the local bus on the IQ80960RM/RN platform.

3.2.1 SDRAM Performance

The IQ80960RM/RN platform uses 72-bit SDRAM with ECC or 64-bit SDRAM without ECC. SDRAM allows zero data-to-data wait state operation at 66 MHz. The memory controller unit (MCU) of the i960® RM/RN I/O processor supports SDRAM burst lengths of four. A burst length of four enables seamless read/write bursting of long data streams, as long as the MCU does not cross the page boundary. Page boundaries are naturally aligned 2 Kbyte blocks. 72-bit SDRAM with ECC allows a maximum throughput of 528 Mbytes per second.

Both 16 Mbit and 64 Mbit SDRAM devices are supported. The MCU keeps two pages per bank open simultaneously for 16 Mbit devices and 4 pages per bank for 64 Mbit devices. Simultaneously open pages allow for greater performance for sequential access, distributed across multiple internal bus transactions. Table 3-3 shows read and write examples of a single 8 byte access and for a multiple 40 byte access.

Table 3-3. SDRAM Performance

Cycle Type	Table Clocks	Performance Bandwidth
Read Page Hit (8 bytes)	7	76 Mbytes/sec
Read Page Miss (8 bytes)	12	44 Mbytes/sec
Read Page Hit (40 bytes)	11	240 Mbytes/sec
Read Page Miss (40 bytes)	16	165 Mbytes/sec
Write Page Hit (8 bytes)	4	132 Mbytes/sec
Write Page Miss (8 bytes)	8	66 Mbytes/sec
Write Page Hit (40 bytes)	8	330 Mbytes/sec
Write Page Miss (40 bytes)	12	220 Mbytes/sec

Note that if ECC is enabled and you attempt a partial write — less than 64 bits — you will incur a penalty. Because ECC is enabled, the MCU will translate the write into a read-modify-write transaction. Therefore, for a single byte write the clock count will be 11.

3.2.2 Upgrading SDRAM

The IQ80960RM/RN is equipped with 16 Mbytes of SDRAM with ECC inserted in the 168-pin DIMM socket. The memory may be expanded by inserting up to a 128 Mbyte module into the DIMM socket. The various memory combinations are shown in Table 3-4. Only 168-pin +3.3V SDRAM modules with or without ECC rated at 10 ns should be used on the IQ80960RM/RN platform. The column labeled ECC determines if that particular memory configuration can be used with ECC.

Table 3-4. SDRAM Configurations

SDRAM Technology	SDRAM Arrangement	# Banks	Row	Column	ECC	Total Memory SIze
16 Mbit	2M x 8	1	11	9	Yes	16 Mbytes
		2			Yes	32 Mbytes
	1M x 16	1	11	8	No	8 Mbytes
		2			No	16 Mbytes
64 Mbit	8M x 8	1	12	9	Yes	64 Mbytes
		2			Yes	128 Mbytes
	4M x 16	1	12	8	No	32 Mbytes
		2			No	64 Mbytes

3.3 Flash ROM

An E28F016S5 (2 Mbytes) Flash ROM is included on the IQ80960RM/RN platform. This Flash ROM contains IxWorks* and may be used to store user applications.

3.3.1 Flash ROM Programming

Two types of Flash ROM programming exist on the IQ80960RM/RN platform. The first is normal application development programming. This occurs using IxWorks to download new software and the 80960JT core to write the new code to the Flash ROM. During this time the boot sectors (containing IxWorks) are write protected.

The second type of Flash ROM programming is loading the boot sectors. You will not be required to load the boot sectors except:

- To load MON960
- To load a new release of IxWorks
- To change between the check build and the free build of IxWorks

The following steps are required to program the Flash ROM boot sectors:

1. Set switch S1 #'s 1 and 2 to the on position.
2. Reset the board by cycling power on the workstation.
3. Run the Intel DOS-based flash utility to program the Flash ROM boot sectors.
4. Set switch S1 #'s 1 and 2 to the off position.
5. Reset the board by cycling power on the workstation.

3.4 Console Serial Port

The console serial port on the IQ80960RM/RN platform, based on a 16C550 UART, is capable of operation from 300 to 115,200 bps. The port is connected to a phone jack-style plug on the IQ80960RM/RN platform. The DB25 to RJ-45 cable included with the IQ80960RM/RN can be used to connect the console port to any standard RS-232 port on the host system.

The UART on the IQ80960RM/RN platform is clocked with a 1.843 MHz clock, and may be programmed to use this clock with its internal baud rate counters. The UART register addresses are shown in Table 3-5; refer to the 16C550 device data book for a detailed description of the registers and device operation. Note that some UART addresses refer to different registers depending on whether a read or a write is being performed.

Table 3-5. UART Register Addresses

Address	Read Register	Write Register
E000 0000H	Receive Holding Register	Transmit Holding Register
E000 0001H	Unused	Interrupt Enable Register
E000 0002H	Interrupt Status Register	FIFO Control Register
E000 0003H	Unused	Line Control Register
E000 0014H	Unused	Modem Control Register
E000 0015H	Line Status Register	Unused
E000 0016H	Modem Status Register	Unused
E000 0017H	Scratchpad Register	Scratchpad Register

3.5 Secondary PCI Bus Expansion Connectors

Four PCI Expansion Slots are available on the IQ80960RM/RN platform. The IQ80960RM supports 32-bit PCI expansion and the IQ80960RN supports 64-bit PCI expansion. The slots are designed for +5V PCI signalling and accommodate PCI cards with +5V or universal signalling capabilities.

3.5.1 PCI Slots Power Availability

Power from the Primary PCI bus, +3.3V, +5V, +12V, and –12V, is routed to the Secondary PCI bus expansion slots. +3.3V is only available at the secondary PCI slots if the host system makes +3.3V available on the Primary PCI slots. LED CR5 indicates if this power is available.

3.5.2 Interrupt and IDSEL Routing

Table 3-6. Secondary PCI Bus Interrupt and IDSEL Routing

Connector	IDSEL	INTA#	INTB#	INTC#	INTD#
J11	SAD16	SINTA#	SINTB#	SINTC#	SINTD#
J12	SAD17	SINTB#	SINTC#	SINTD#	SINTA#
J13	SAD18	SINTC#	SINTD#	SINTA#	SINTB#
J14	SAD19	SINTD#	SINTA#	SINTB#	SINTC#

3.6 Battery Backup

Battery backup is provided to save any information in SDRAM during a power failure. The IQ80960RM/RN platform contains four AA NiCd batteries, a charging circuit and a regulator circuit. The batteries installed in the IQ80960RM/RN platform are rated at 600 mA/Hr.

SDRAM technology provides a simple way of enabling data preservation though the self-refresh command. When the processor receives an active Primary PCI reset it will issue the self-refresh command and drive the SCKE signals low. Upon seeing this condition a PAL on the IQ80960RM/RN platform will hold SCKE low before the processor loses power. The batteries will maintain power to the SDRAM and the PAL to ensure self-refresh mode. When the PAL sees PRST# returning to inactive state the PAL will release the hold on SCKE.

The battery circuit can be disabled by removing the batteries. LED CR4 indicates when the SDRAMs have sufficient power. If the batteries remain in the evaluation platform when it is depowered and/or removed from the chassis, the batteries will maintain the SDRAM for approximately 30 hours. Once power is again applied, the batteries will be fully charged in about four hours.

3.7 Loss of Fan Detect

The i960 RM/RN I/O processor can be cooled by an active heat sink mounted on top. The fan provides a square wave output that is monitored by a comparator circuit on the IQ80960RM/RN platform. The frequency of the fan output is approximately 9K RPM. If the frequency falls below approximately 8K RPM the circuit will provide an interrupt to the processor. This is an evaluation board feature intended as an example of system hardware monitoring, since the IQ80960RM/RN platform does not ship with a heatsink.

Note: When using a passive heat sink, the processor never sees an interrupt from not having a fan.

3.8 Logic Analyzer Headers

There are five logic analyzer connectors on the IQ80960RM/RN platform. The connectors are Mictor type, AMP part # 767054-1. Hewlett-Packard and Tektronix manufacture and sell interfaces to these connectors. The logic analyzer connectors allow for interfacing to the SDRAM and ROM buses along with secondary PCI arbitration signals. Table 3-7 shows the connectors and the pin assignments for each.

Table 3-7. Logic Analyzer Header Definitions

PIN	J9	J11	J12	J10	J8
3		SDRAMCLK			
4	DQ15	SDQM7	DQ31		RAD15
5	DQ14	SDQM6	DQ30		RAD14
6	DQ13	SDQM5	DQ29		RAD13
7	DQ12	SDQM4	DQ28		RAD12
8	DQ11	SDQM3	DQ27		RAD11
9	DQ10	SDQM2	DQ26		RAD10
10	DQ9	SDQM1	DQ25		RAD9
11	DQ8	SDQM0	DQ24		RAD8
12	DQ7	SCB7	DQ23		RAD7
13	DQ6	SCB6	DQ22		RAD6
14	DQ5	SCB5	DQ21		RAD5
15	DQ4	SCB4	DQ20		RAD4
16	DQ3	SCB3	DQ19	SCE0#	RAD3
17	DQ2	SCB2	DQ18	SCE1#	RAD2
18	DQ1	SCB1	DQ17	SBA1	RAD1
19	DQ0	SCB0	DQ16	SBA0	RAD0
20	DQ32	SA0	DQ48	SREQ0#	RAD16
21	DQ33	SA1	DQ49	SREQ1#	
22	DQ34	SA2	DQ50	SREQ2#	
23	DQ35	SA3	DQ51	SREQ3#	RALE
24	DQ36	SA4	DQ52	SREQ4#	RCE0#
25	DQ37	SA5	DQ53	SREQ5#	RCE1#
26	DQ38	SA6	DQ54	SGNT0#	ROE#
27	DQ39	SA7	DQ55	SGNT1#	RWE#
28	DQ40	SA8	DQ56	SGNT2#	
29	DQ41	SA9	DQ57	SGNT3#	I_RST#
30	DQ42	SA10	DQ58	SGNT4#	
31	DQ43	SA11	DQ59	SGNT5#	
32	DQ44		DQ60		
33	DQ45	SWE#	DQ61		
34	DQ46	SCAS#	DQ62		
35	DQ47	SRAS#	DQ63		
36				P_PCICLK	RALE

3.9 JTAG Header

The JTAG header allows debugging hardware to be quickly and easily connected to some of the IQ80960RM/RN processor's logic signals.

The JTAG header is a 16-pin header. A 3M connector (part number 2516-6002UG) is required to connect to this header. The pinout for the JTAG header is shown in Table 3-8. The header and connector are keyed using a tab on the connector and a slot on the header to ensure proper installation.

Each signal in the JTAG header is paired with its own ground connection to avoid the noise problems associated with long ribbon cables. Signal descriptions are found in the *i960® RM/RN I/O Processor Developer's Manual*, *80960RM I/O Processor* Data Sheet and the *80960RN I/O Processor* Data Sheet.

Table 3-8. JTAG Header Pinout

Pin	Signal	Input/Output to 80960RM/RN	Pin	Signal
1	TRST#	IN	2	GND
3	TDI	IN	4	GND
5	TDO	OUT	6	GND
7	TMS	IN	8	GND
9	TCK	IN	10	GND
11	LCDINIT#	IN	12	GND
13	I_RST#	OUT	14	GND
15	PWRVLD	OUT	16	GND

Table 3-9 describes switch setting options and defaults. These switch settings are sampled at Primary PCI Reset. See Table 5-1 "Initialization Modes" on page 5-3 for processor initialization configurations.

Table 3-9. Switch S1 Settings

Position	Name	Description	Default
S1-1	RST_MODE#	Determines if the processor is to be held in reset. ON = hold in rest OFF = allows processor initialization	OFF
S1-2	RETRY	Determines if the Primary PCI interface will be disabled. ON = allows Primary PCI configuration cycles to occur OFF = retries all Primary PCI configuration cycles	OFF
S1-3	32BITMEM_EN#	Notifies Memory Controller of the SDRAM width. ON = Memory Controller utilizes 32-bit SDRAM access protocol OFF = Memory Contoller utilizes 64-bit SDRAM access protocol	OFF
S1-4[a]	32BITPCI_EN#	Determines whether Secondary PCI bus is a 32- or 64-bit bus. ON = indicates Secondary PCI bus is a 32-bit bus OFF = indicates Secondary PCI bus is a 64-bit bus	OFF

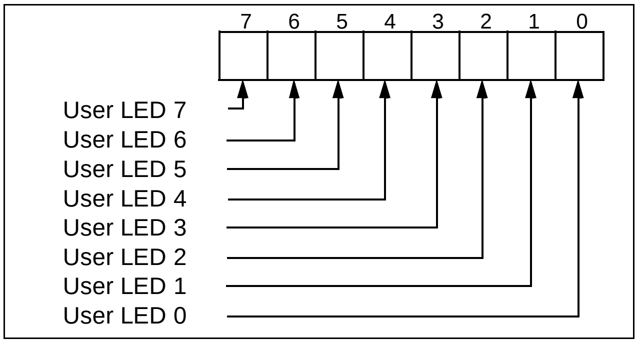
a. This switch is active for IQ80960RN ONLY.

3.10 User LEDs

The IQ80960RM/RN platform has a bank of eight user-programmable LEDs, located on the upper edge of the adapter board. These LEDs are controlled by a write-only register and used as a debugging aid during development. Software can control the state of the user LEDs by writing to the LED Register, located at E004 0000H. Each of the eight bits of this register correspond to one of the user LEDs. Clearing a bit in the LED Register by writing a "0" to it turns the corresponding LED "on", while setting a bit by writing a "1" to it turns the corresponding LED "off". Resetting the IQ80960RM/RN platform results in clearing the register and turning all the LEDs "on". The LED Register bitmap is shown in Figure 3-1.

The user LEDs are numbered in descending order from left to right, with LED7 being on the left when looking at the component side of the adapter.

Figure 3-1. LED Register Bitmap



3.10.1 User LEDs During Initialization

MON960 indicates the progress of its hardware initialization on the user LEDs. In the event that initialization should fail for some reason, the number of lit LEDs can be used to determine the cause of the failure. Table 3-10 lists the tests that correspond to each lit LED.

Table 3-10. Start-up LEDs MON960

LEDs	Tests
LED 0	SDRAM serial EEPROM checksum validated
LED 1	UART walking ones test passed
LED 2	DRAM walking ones test passed
LED 3	DRAM multiword test passed
LED 4	Hardware initialization started
LED 5	Flash ROM initialized
LED 6	PCI-to-PCI Bridge initialized
LED 7	UART internal loopback test passed

Table 3-11 lists the connectors and LEDs.

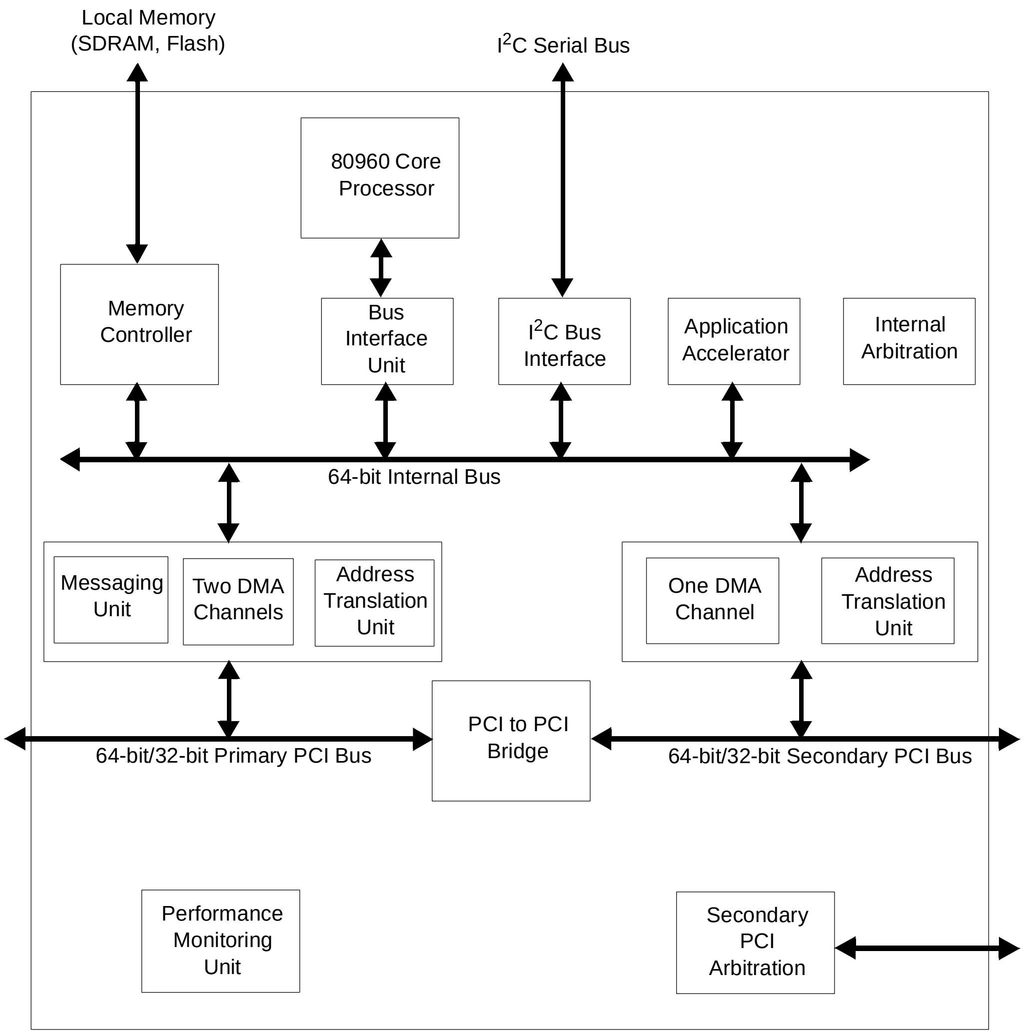
Table 3-11. IQ80960RM/RN Connectors and LEDs

Item	Description
J1-J4	Secondary PCI bus expansion connector
J5	168-pin SDRAM DIMM socket
J6	JTAG connector
J7	Serial port connector
J8	Logic analyzer connector for flash ROM bus
J10	Logic analyzer connector for Secondary PCI bus arbitration signals
J9, J11, J12	Logic analyzer connector for access to SDRAM bus
J13	Active heatsink connector for example fan monitor circuit
CR1, CR2	Eight user LEDs
CR3	Self-test fail LED
CR4	Battery backup SDRAM, 3.3 V available
CR5	Indicates host system providing 3.3 V to Secondary PCI bus connectors
S1	DIP switch (Table 3-9)

i960® RM/RN I/O Processor Overview 4

This chapter describes the features and operation of the processor on the IQ80960RM/RN platform. For more detail, refer to the *i960® RM/RN I/O Processor Developer's Manual*.

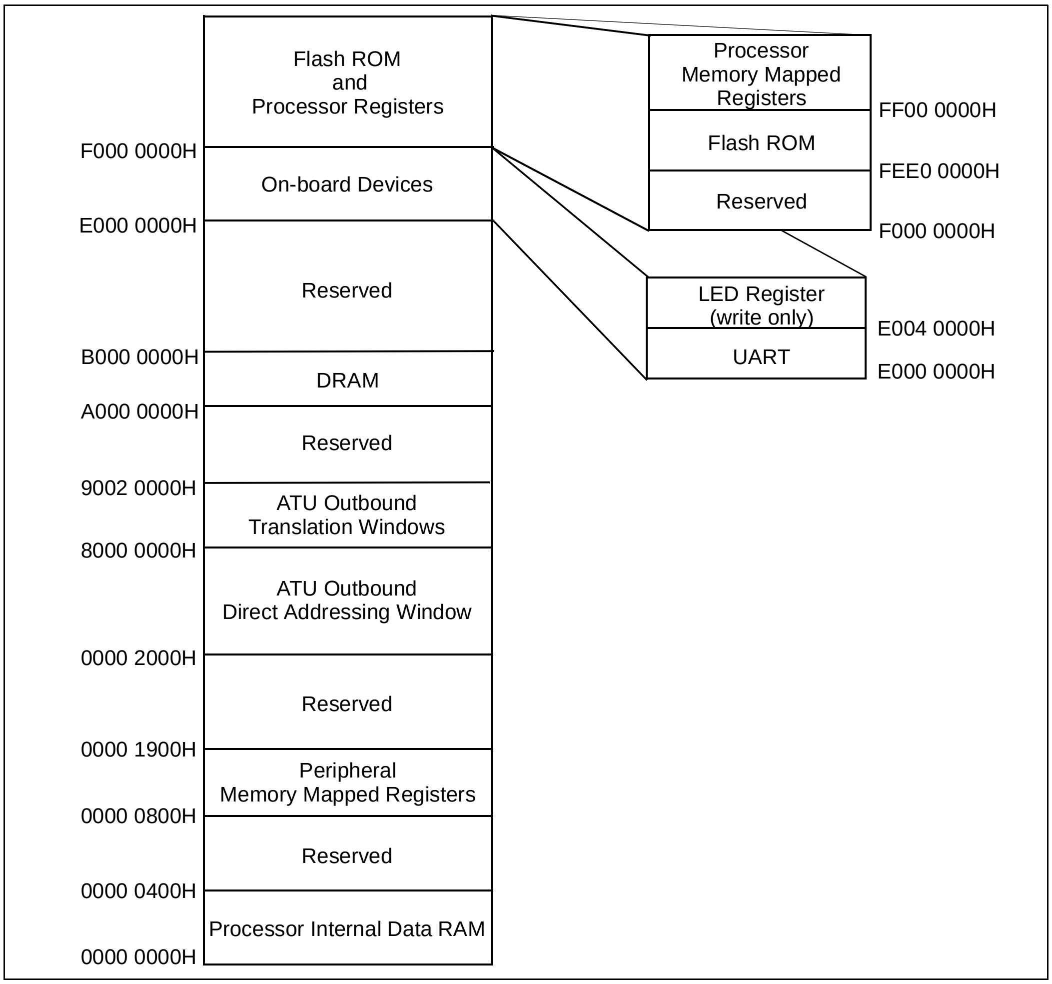
Figure 4-1. i960® RM/RN I/O Processor Block Diagram



4.1 CPU Memory Map

The memory map for the IQ80960RM/RN platform is shown in Figure 4-2. All addresses below 9002 0000H on the IQ80960RM/RN platform are reserved for various functions of the i960 RM/RN I/O processor, as shown on the memory map. Documentation for these areas, as well as the processor memory mapped registers at FF00 0000H and the IBR, can be found in the *i960® RM/RN I/O Processor Developer's Manual*.

Figure 4-2. IQ80960RM/RN Platform Memory Map



4.2 Local Interrupts

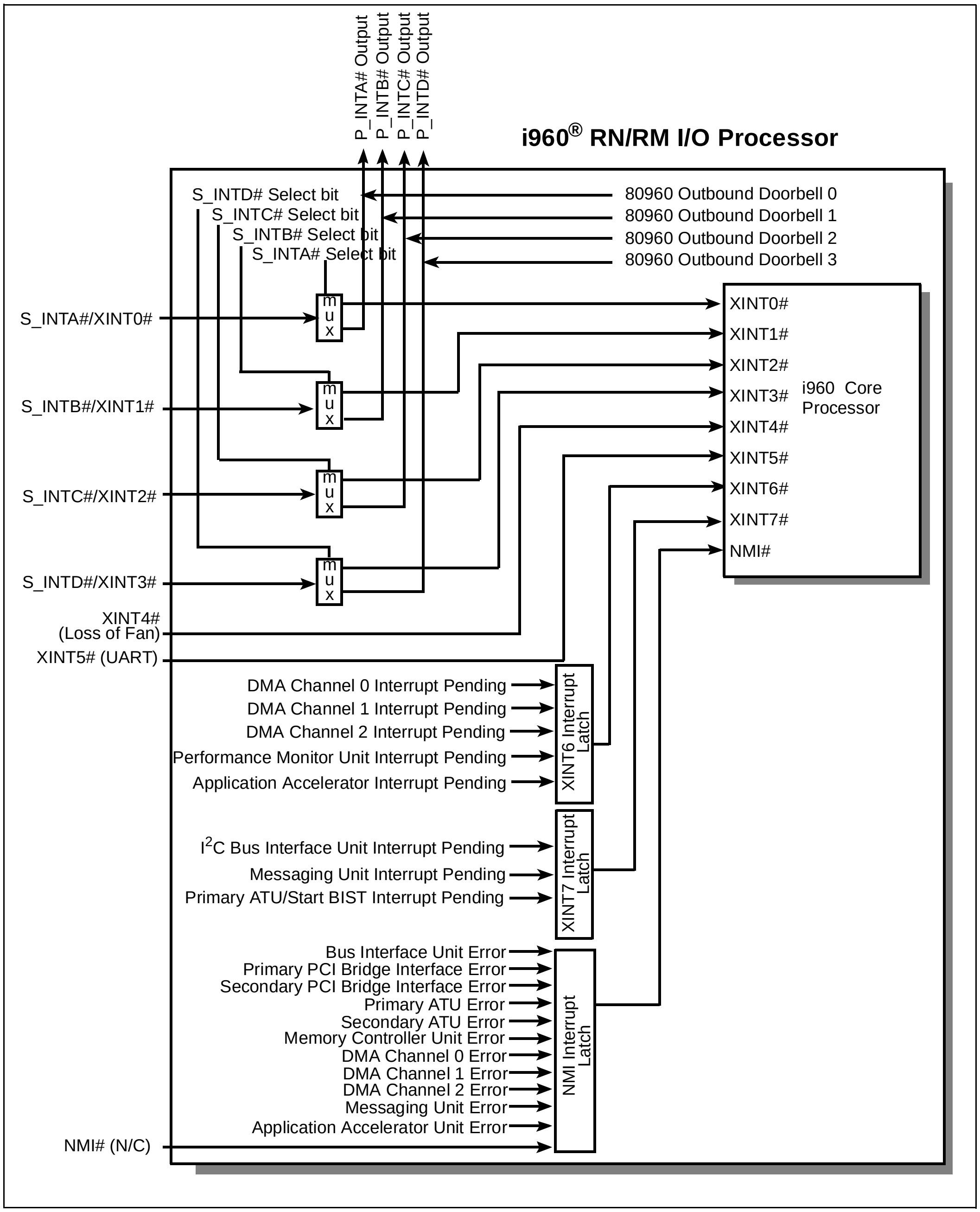
The i960 RM/RN I/O processor is built around an 80960JT core, which has seven external interrupt lines designated XINT0# through XINT5# and NMI#. In the i960 RM/RN I/O processor, these interrupt lines are not directly connected to external interrupts, but pass through a layer of internal interrupt routing logic. Figure 4-3 shows the interrupt connections on the i960 RM/RN I/O processor.

XINT0# through XINT3# on the 80960JT core can be used to receive PCI interrupts from the secondary PCI bus, or these interrupts can be passed through to the primary PCI interface, depending on the setting of the XINT Select bit of the PCI Interrupt Routing Select Register in the i960 RM/RN I/O processor. On the IQ80960RM/RN platform, XINT0# through XINT3# are configured to receive interrupts from the secondary PCI bus.

XINT4# and XINT5# on the i960 RM/RN I/O processor may be connected to interrupt sources external to the processor. On the IQ80960RM/RN platform, XINT4# is connected to the loss of fan detect and XINT5# is connected to the 16C550 UART.

XINT6#, XINT7# receive interrupts from internal sources. NMI# receives interrupts from internal sources and from an external source. Since all of these interrupts accept signals from multiple sources, a status register is provided for each of them to allow service routines to identify the source of the interrupt. Each of the possible interrupt sources is assigned a bit position in the status register. The interrupt sources for these lines are shown in Figure 4-3. On the IQ80960RM/RN platform, the NMI# interrupt is not connected to any external interrupt source and receives interrupts only from the internal devices on the i960 RM/RN I/O processor. Note that all error conditions result in an NMI# interrupt.

Figure 4-3. i960® RM/RN I/O Processor Interrupt Controller Connections



4.3 CPU Counter/Timers

The i960 RM/RN I/O processor is equipped with two on-chip counter/timers which are clocked with the i960 RM/RN I/O processor clock signal. The i960 RM/RN I/O processor receives its clock from the primary PCI interface clock, generated by the motherboard. Most motherboards generate a 33 MHz clock signal, although the PCI specification requires a clock frequency between 0 and 33 MHz. The timers can be programmed for single-shot or continuous mode, and can generate interrupts to the processor when the countdown expires.

4.4 Primary PCI Interface

The primary PCI interface on the IQ80960RM/RN platform provides the i960 RM/RN I/O processor with a connection to the PCI bus on the host system. Only the PCI-to-PCI bridge unit on the i960 RM/RN I/O processor is directly connected to the primary PCI interface. Devices installed on the expansion slots are connected to the PCI bus via the bridge unit on the i960 RM/RN I/O processor. The PCI-to-PCI bridge accepts Type 1 configuration cycles destined for devices on the secondary bus, and will forward them as Type 0 or Type 1 configuration cycles, or as special cycles. The IQ80960RN platform interfaces to a 64-bit PCI bus and the IQ80960RM platform interfaces to a 32-bit PCI bus.

4.5 Secondary PCI Interface

The secondary PCI interface provided by the i960 RM/RN I/O processor is used to connect PCI cards via the expansion slots to the host system’s PCI bus. PCI cards are attached to the IQ80960RM/RN platform with a standard PCI connector and may contain up to four separate PCI devices. The i960 RM/RN I/O processor provides PCI-to-PCI bridge functionality to map installed PCI devices onto the host PCI bus, and supports transaction forwarding in both directions across the bridge. PCI devices connected via the expansion slots can therefore act as masters or slaves on the host system’s PCI bus. Additional PCI-to-PCI bridge devices are supported by the i960 RM/RN I/O processor on its secondary PCI interface and can be designed into add-on PCI cards. In addition, the i960 RM/RN I/O processor supports “private” PCI devices on its secondary bus. Private devices are hidden from initialization code on the host system, and are configured and accessed directly by the i960 RM/RN I/O processor. These devices are not part of the normal PCI address space, but they can act as PCI bus masters and transfer data to and from other PCI devices in the system.

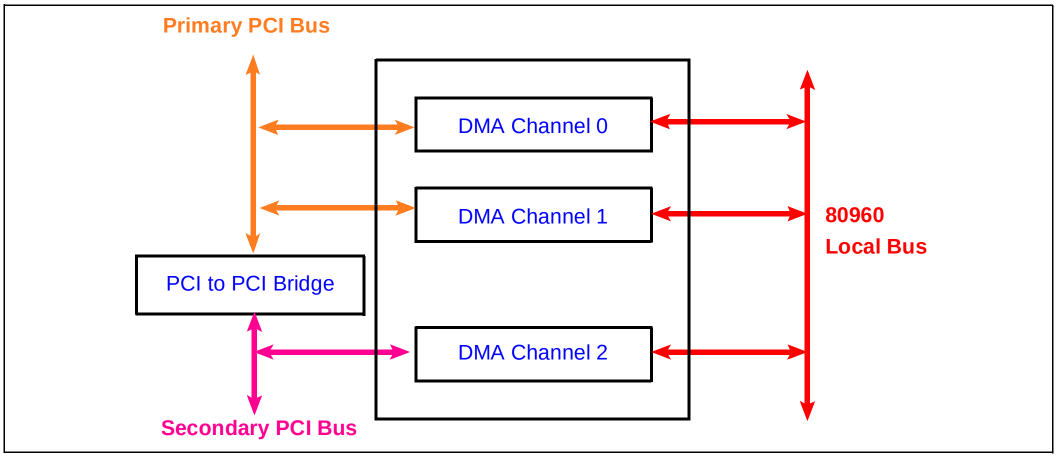
Unless designated as private devices, PCI devices installed on the secondary PCI interface of the IQ80960RM/RN platform are mapped into the system-wide PCI address space by configuration software running on the host system. No logical distinction is made at the system level between devices on the primary PCI bus and devices on secondary buses; all transaction forwarding is handled transparently by the PCI-to-PCI bridge. Configuration cycles and read and write accesses from the host are forwarded through the PCI-to-PCI bridge unit of the i960 RM/RN I/O processor. Master read and write cycles from devices on the secondary PCI bus are also forwarded to the host bus by the PCI-to-PCI bridge unit.

IxWORKS allows secondary PCI devices to be configured as Public or Private. Public devices are configured by the PCI host. Private devices are configured by the IxWORKS kernel and the device-specific HDM.

4.6 DMA Channels

The i960 RM/RN I/O processor features three independent DMA channels, two of which operate on the primary PCI interface, whereas the remaining one operates on the secondary PCI interface. All three of the DMA channels connect to the i960 RM/RN I/O processor's local bus and can be used to transfer data from PCI devices to memory on the IQ80960RM/RN platform. Support for chaining, and scatter/gather is built into all three channels. The DMA can address the entire 2^{64} bytes of address space on the PCI bus and 2^{32} bytes of address space on the internal bus.

Figure 4-4. i960® RM/RN I/O Processor DMA Controller



4.7 Application Accelerator Unit

The Application Accelerator provides low-latency, high-throughput data transfer capability between the AA unit and 80960 local memory. It executes data transfers to and from 80960 local memory and also provides the necessary programming interface. The Application Accelerator performs the following functions:

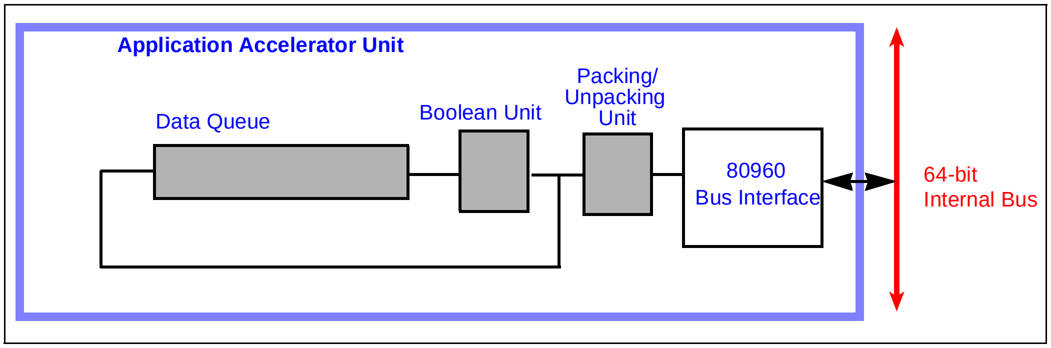
- Transfers data (read) from memory controller
- Performs an optional boolean operation (XOR) on read data
- Transfers data (write) to memory controller

The AA unit features:

- 128-byte, arranged as 8-byte x 16-deep store queue
- Utilization of the 80960RN/RM processor memory controller interface
- 2^{32} addressing range on the 80960 local memory interface
- Hardware support for unaligned data transfers for the internal bus
- Full programmability from the i960 core processor
- Support for automatic data chaining for gathering and scattering of data blocks

Figure 4-5 shows a simplified connection of the Application Accelerator to the i960 RM/RN I/O Processor Internal Bus.

Figure 4-5. Application Accelerator Unit



4.8 Performance Monitor Unit

The Performance Monitoring features aid in measuring and monitoring various system parameters that contribute to the overall performance of the processor. The monitoring facility is generically referred to as PMON – Performance Monitoring. The facility is model specific, not architectural; its intended use is to gather performance measurements that can be used to retune/refine code for better system level performance.

The PMON facility provided on the i960 RM/RN I/O processor comprises:

- One dedicated global Time Stamp counter, and
- Fourteen (14) Programmable Event counters

The global time stamp counter is a dedicated, free running 32-bit counter.

The programmable event counters are 32-bits wide. Each counter can be programmed to observe an event from a defined set of events. An event consists of a set of parameters which define a start condition and a stop condition. The monitored events are selected by programming an event select register (ESR).

MON960 Support for IQ80960RM/RN 5

This chapter discusses a number of additions that have been made to MON960 to support the IQ80960RM/RN in an optional non-I^2O capacity. For complete documentation on the operation of MON960, see the *MON960 Debug Monitor User's Guide*. The IQ80960RM/RN evaluation platform ships with IxWorks* from Wind River Systems installed in flash firmware. To use CTOOLS and MON960 instead of IxWorks, you need to download MON960 into the onboard Flash. See Chapter 2 for more information on updating the onboard Flash. See Chapter 1 for descriptions of both IxWorks and CTOOLS.

5.1 Secondary PCI Bus Expansion Connectors

The IQ80960RM/RN platform contains four secondary PCI bus expansion connectors to give users access to the secondary PCI bus of the i960® RM/RN I/O processor. Extensions to MON960 perform secondary PCI bus initialization including the establishment of a secondary PCI bus address map. Routines compatible with the *PCI Local Bus Specification* Revision 2.1 allow the software on the IQ80960RM/RN platform to search for devices on the secondary PCI bus and read and write the configuration space of those devices.

5.2 MON960 Components

The remaining sections of this chapter assume that MON960 is installed in the onboard Flash, replacing IxWorks. The IQ80960RM/RN optional MON960 debug monitor consists of four main components:

- Initialization firmware
- MON960 kernel
- MON960 extensions
- Diagnostics/example code

These four components together are referred to as MON960.

5.2.1 MON960 Initialization

At initialization, MON960 puts the IQ80960RM/RN platform into a known, functional state that allows the host processor to perform PCI initialization. Once in this state, the MON960 kernel and the MON960 extensions can load and execute correctly. Initialization is performed after a RESET condition. MON960 initialization encompasses all major portions of the i960 RM/RN I/O processor and IQ80960RM/RN platform including 80960JT core initialization, Memory Controller initialization, SDRAM initialization, Primary PCI Address Translation Unit (ATU) initialization, and PCI-to-PCI Bridge Unit initialization.

The IQ80960RM/RN platform is designed to use the Configuration Mode of the i960 RM/RN I/O processor. Configuration Mode allows the 80960JT core to initialize and control the initialization process before the PCI host configures the i960 RM/RN I/O processor. By utilizing Configuration Mode, the user

is given the ability to initialize the PCI configuration registers to values other than the default power-up values. Configuration Mode gives the user maximum flexibility to customize the way in which the i960 RM/RN I/O processor and IQ80960RM/RN platform appear to the PCI host configuration software.

5.2.2 80960JT Core Initialization

The 80960JT core begins the initialization process by reading its Initial Memory Image (IMI) from a fixed address in the boot ROM (FEFF FF30H in the i960 address space). The IMI includes the Initialization Boot Record (IBR), the Process Control Block (PRCB), and several system data structures. The IBR provides initial configuration information for the core and integrated peripherals, pointers to the system data structures and the first instruction to be executed after processor initialization, and checksum words that the processor uses in its self-test routine. In addition to the IBR and PRCB, the required data structures are the:

- System Procedure Table
- Control Table
- Interrupt Table
- Fault Table
- User Stack (application dependent)
- Supervisor Stack
- Interrupt Stack

5.2.3 Memory Controller Initialization

Since the i960 RM/RN I/O processor Memory Controller is integral to the design and operation of the IQ80960RM/RN platform, the operational parameters for Bank 0 and Bank 1 are established immediately after processor core initialization. Memory Bank 0 is associated with the ROM on the IQ80960RM/RN platform. Memory Bank 1 is associated with the UART and the LED Control Register. Parameters such as Bank Base Address, Read Wait States, and Write Wait States must be established to ensure the proper operation of the IQ80960RM/RN platform. The Memory Controller is initialized so as to be consistent with the IQ80960RM/RN platform memory map shown in Figure 4-2.

5.2.4 SDRAM Initialization

SDRAM initialization includes setting operational parameters for the SDRAM controller, and sizing and clearing the installed SDRAM configuration. To configure the system properly, Presence Detect data is read from the EEPROM of the SDRAM module, using the 80960RM/RN I^2C Bus Interface Unit. Presence Detect data includes the number and size of SDRAM banks present on the installed module. On power-up, 64 bytes of Presence Detect data are read and validated. The SDRAM controller is then configured by setting the base address of SDRAM, the boundary limits for each SDRAM bank, the refresh cycle interval, and the output buffer drive strength. Once the SDRAM controller is configured, the SDRAM is cleared in preparation for the C language runtime environment. The actual SDRAM size is stored for later use (e.g., to establish the size of the IQ80960RM/RN platform PCI Slave image). The SDRAM controller is initialized to be consistent with the IQ80960RM/RN platform memory map shown in Figure 4-2.

5.2.5 Primary PCI Interface Initialization

The IQ80960RM/RN platform is a multi-function PCI device. On the primary PCI bus, two functions (from a PCI Configuration Space standpoint) are supported.

- Function 0 is the PCI-to-PCI Bridge of the i960 RM/RN I/O processor, which optionally provides access capability between the primary PCI bus and the secondary PCI bus.
- Function 1 is the Primary ATU which provides access capability between the primary PCI bus and the local i960 bus.

The platform can be initialized into one of four modes. Modes 0 and 3 are described below.

Table 5-1. Initialization Modes

RST_MODE#/ SW1-1	RETRY/ SW1-2	Initialization Mode	Primary PCI Interface	i960 Core Processor
0/ON	0/ON	Mode 0	Accepts Transactions	Held in Reset
0/ON	1/OFF	Mode 1	Retries All Configuration Transactions	Held in Reset
1/OFF	0/ON	Mode 2	Accepts Transactions	Initializes
1/OFF	1/OFF	Mode 3 (default)	Retries All Configuration Transactions	Initializes

When the IQ80960RM/RN is operating in Mode 0, the processor core is held in reset, allowing register defaults to be used on the Primary PCI interface. This mode is used to program the onboard Flash with either IxWORKS* or MON960.

When the IQ80960RM/RN platform is operating in Mode 3, the Configuration Cycle Disable bit in the Extended Bridge Control Register (EBCR) is set after IQ80960RM/RN processor reset. In this mode, the IQ80960RM/RN platform sends PCI Retries when the PCI host attempts to access the platform's Configuration Space. This mode allows the IQ80960RM/RN processor time to initialize its internal registers. The processor remains in this mode until the Configuration Cycle Disable bit in the Extended Bridge Control Register (EBCR) is cleared. For this reason, and to prevent PCI host problems, Primary PCI initialization occurs at the earliest possible opportunity after Memory and SDRAM controller initialization.

5.2.6 Primary ATU Initialization

Primary ATU (Bridge) initialization includes initialization by the 80960JT core and initialization by the PCI host processor. Local initialization occurs first and consists mainly of establishing the operational parameters for access to the local IQ80960RM/RN platform bus. The Primary Inbound ATU Limit Register (PIALR) is initialized to establish the block size of memory required by the Primary ATU. The PIALR value is based on the installed SDRAM configuration. The Primary Inbound ATU Translate Value Register (PIATVR) is initialized to establish the translation value for PCI-to-Local accesses. The PIATVR value is set to reference the base of local SDRAM. The Primary Outbound Memory Window Value Register (POMWVR) is initialized to establish the translation value for Local-to-PCI accesses. The POMWVR value remains at its default value of "0" to allow the IQ80960RM/RN platform to access the start of the PCI Memory address map, which is typically occupied by PCI host memory. Likewise, the Primary Outbound I/O Window Value Register (POIOWVR) remains at its default value of "0" to allow the IQ80960RM/RN platform to access the start of the PCI I/O address map. PCI Doorbell-related parameters are also established to allow for communication between the IQ80960RM/RN platform and a PCI bus master using the doorbell mechanism.

By default, Primary Outbound Configuration Cycle parameters are not established. The ATU Configuration Register (ATUCR) is initialized to establish the operational parameters for the Doorbell Unit and ATU interrupts (both primary and secondary), and to enable the primary and secondary ATUs. The PCI host is responsible for allocating PCI address space (Memory, Memory Mapped I/O, and I/O), and assigning the PCI Base addresses for the IQ80960RM/RN platform.

5.2.7 PCI-to-PCI Bridge Initialization

PCI-to-PCI Bridge initialization includes initialization by the 80960JT core and initialization by the PCI host processor. Local initialization occurs first and consists mainly of establishing the operational parameters for the secondary PCI interface of the PCI-to-PCI bridge. On the IQ80960RM/RN platform, the secondary PCI bus is configured to consist of private devices (not visible to PCI host configuration cycles). To support a private secondary PCI bus, the Secondary IDSEL Select Register (SISR) is initialized to prevent the secondary PCI address bits [20:16] from being asserted during conversion of PCI Type 1 configuration cycles on the primary PCI bus to PCI Type 0 configuration cycles on the secondary PCI bus. Secondary PCI bus masters are prevented from initiating transactions that will be forwarded to the primary PCI interface. The PCI host is responsible for assigning and initializing the PCI bus numbers, allocating PCI address space (Memory, Memory Mapped I/O, and I/O), and assigning the IRQ numbers to valid interrupt routing values.

5.2.8 Secondary ATU Initialization

Secondary ATU (Bridge) initialization consists mainly of establishing the operational parameters for access between the local IQ80960RM/RN platform bus and the secondary PCI devices. The Secondary Inbound ATU Base Address Register (SIABAR) is initialized to establish the PCI base address of IQ80960RM/RN platform local memory from the secondary PCI bus. By convention, the secondary PCI base address for access to IQ80960RM/RN platform local memory is “0”. The Secondary Inbound ATU Limit Register (SIALR) is initialized to establish the block size of memory required by the secondary ATU. The SIALR value is based on the installed SDRAM configuration. The Secondary Inbound ATU Translate Value Register (SIATVR) is initialized to establish the translation value for Secondary PCI-to-Local accesses. The SIATVR value is set to reference the base of local SDRAM. The Secondary Outbound Memory Window Value Register (SOMWVR) is initialized to establish the translation value for Local-to-Secondary PCI accesses. The SOMWVR value is left at its default value of “0” to allow the IQ80960RM/RN platform to access the start of the PCI Memory address map. Likewise, the Secondary Outbound I/O Window Value Register (SOIOWVR) is left at its default value of “0” to allow the IQ80960RM/RN platform to access the start of the PCI I/O address map.

On the secondary PCI bus, the IQ80960RM/RN platform assumes the duties of PCI host and, as such, is required to configure the devices of the secondary PCI bus. Secondary Outbound Configuration Cycle parameters are established during secondary PCI bus configuration. Secondary PCI bus configuration is accomplished via MON960 Extension routines.

5.3 MON960 Kernel

The MON960 Kernel (monitor) provides the IQ80960RM/RN user with a software platform on which application software can be developed and run. The monitor provides several features available to the IQ80960RM/RN user to speed application development. Among the available features are:

- Communication with a terminal or terminal emulation package on a host computer through a serial cable with automatic baud rate detection
- Communication with a software debugger such as GDB960 (available from Intel) using the Host Debugger Interface (HDI) software interface
- Communication with the host computer via the primary PCI bus
- Downloads of ELF object files via the primary PCI bus or via the serial console port at rates up to 115,200 baud
- Downloads of ELF object files via the primary PCI bus
- On-board erasure and programming of Intel 28F016S5 Flash ROM
- Memory display and modification capability
- Breakpoint and single-step capability to support debugging of user code
- Disassembly of i960 processor instructions

5.4 MON960 Extensions

The monitor has been extended to include the secondary PCI bus initialization and also the BIOS routines which are contained in the *PCI BIOS Specification* Revision 2.1.

5.4.1 Secondary PCI Initialization

MON960 extensions are responsible for initializing the devices on the secondary PCI bus of the IQ80960RM/RN platform. Secondary PCI initialization involves allocating address spaces (Memory, Memory Mapped I/O, and I/O), assigning PCI base addresses, assigning IRQ values, and enabling PCI mastership. MON960 does not support devices containing PCI-to-PCI bridges and hierarchical buses.

5.4.2 PCI BIOS Routines

MON960 includes PCI BIOS routines to aid application software initialization of the secondary PCI bus. The supported BIOS functions are described in the subsections that follow.

sysPCIBIOSPresent
sysFindPCIDevice
sysFINDPCIClassCode
sysGenerateSpecialCycle
sysReadConfigByte
sysReadConfigWord
sysReadConfigDword
sysWriteConfigByte
sysWriteConfigWord
sysWriteConfigDword
sysGetIrqRoutingOptions
sysSetPCIIrq

These functions preserve, as closely as possible, the parameters and return values described in the *PCI Local Bus Specification* Revision 2.1. Functions that return multiple values do so by filling in the fields of a structure passed by the calling routine.

You can access these functions via a `calls` instruction. The system call indices are defined in the MON960 source file `PCI_BIOS.H`. The function prototypes are defined in the `IQRP_ASM.H` file.

5.4.2.1 sysPCIBIOSPresent

This function allows the caller to determine whether the PCI BIOS interface function set is present, and the current interface version level. It also provides information about the hardware mechanism used for accessing configuration space and whether or not the hardware supports generation of PCI Special Cycles.

Calling convention:

```
int sysPCIBIOSPresent (
    PCI_BIOS_INFO *info
    );
```

Return values:

This function always returns SUCCESSFUL.

5.4.2.2 sysFindPCIDevice

This function returns the location of PCI devices that have a specific Device ID and Vendor ID. Given a Vendor ID, a Device ID, and an Index, the function returns the Bus Number, Device Number, and Function Number of the Nth Device/Function whose Vendor ID and Device ID match the input parameters.

Calling software can find all devices having the same Vendor ID and Device ID by making successive calls to this function starting with the index set to “0”, and incrementing the index until the function returns DEVICE_NOT_FOUND. A return value of BAD_VENDOR_ID indicates that the Vendor ID value passed had a value of all “1”s.

Calling convention:

```
int sysFindPCIDevice (
    int   device_id,
    int   vendor_id,
    int   index
    );
```

Return values:

This function returns SUCCESSFUL if the indicated device is located, DEVICE_NOT_FOUND if the indicated device cannot be located, or BAD_VENDOR_ID if the vendor_id value is illegal.

5.4.2.3 sysFindPCIClassCode

This function returns the location of PCI devices that have a specific Class Code. Given a Class Code and an Index, the function returns the Bus Number, Device Number, and Function Number of the Nth Device/Function whose Class Code matches the input parameters.

Calling software can find all devices having the same Class Code by making successive calls to this function starting with the index set to “0”, and incrementing the index until the function returns DEVICE_NOT_FOUND.

Calling convention:

```
int sysFindPCIClassCode (
        int     class_code,
        int     index
        );
```

Return values:

This function returns SUCCESSFUL when the indicated device is located, or DEVICE_NOT_FOUND when the indicated device cannot be located.

5.4.2.4 sysGenerateSpecialCycle

This function allows for generation of PCI Special Cycles. The generated special cycle is broadcast on a specific PCI Bus in the system.

PCI Special Cycles are not supported on the IQ80960RM/RN platform secondary PCI bus.

Calling convention:

```
int sysGenerateSpecialCycle (
        int bus_number,
        int special_cycle_data
        );
```

Return values:

Since PCI Special Cycles are not supported by the IQ80960RM/RN platform, this function always returns FUNC_NOT_SUPPORTED.

5.4.2.5 sysReadConfigByte

This function allows the caller to read individual bytes from the configuration space of a specific device.

Calling convention:

```
int sysReadConfigByte (
        int     bus_number,
        int     device_number,
        int     function_number,
        int     register_number,          /* 0,1,2,...,255 */
        UINT8       *data
        );
```

Return values:

This function returns SUCCESSFUL when the indicated byte was read correctly, or ERROR when there is a problem with the parameters.

5.4.2.6 sysReadConfigWord

This function allows the caller to read individual shorts (16 bits) from the configuration space of a specific device. The Register Number parameter must be a multiple of two (i.e., bit 0 must be set to "0").

Calling convention:

```
int sysReadConfigWord (
        int     bus_number,
        int     device_number,
        int     function_number,
        int     register_number,            /* 0,2,4,...,254 */
        UINT16          *data
        );
```

Return values:

This function returns SUCCESSFUL when the indicated word was read correctly, or ERROR when there is a problem with the parameters.

5.4.2.7 sysReadConfigDword

This function allows the caller to read individual longs (32 bits) from the configuration space of a specific device. The Register Number parameter must be a multiple of four (i.e., bits 0 and 1 must be set to "0").

Calling convention:

```
int sysReadConfigDword (
        int     bus_number,
        int     device_number,
        int     function_number,
        int     register_number,            /* 0,4,8,...,252 */
        UINT32 *data
        );
```

Return values:

This function returns SUCCESSFUL when the indicated long was read correctly, or ERROR when there is a problem with the parameters.

5.4.2.8 sysWriteConfigByte

This function allows the caller to write individual bytes to the configuration space of a specific device.

Calling convention:

```
int sysWriteConfigByte (
        int     bus_number,
        int     device_number,
        int     function_number,
        int     register_number,          /* 0,1,2,...,255 */
        UINT8        *data
        );
```

Return values:

This function returns SUCCESSFUL when the indicated byte was written correctly, or ERROR when there is a problem with the parameters.

5.4.2.9 sysWriteConfigWord

This function allows the caller to write individual shorts (16 bits) to the configuration space of a specific device. The Register Number parameter must be a multiple of two (i.e., bit 0 must be set to “0”).

Calling convention:

```
int sysWriteConfigWord (
        int     bus_number,
        int     device_number,
        int     function_number,
        int     register_number,          /* 0,2,4,...,254 */
        UINT16       *data
        );
```

Return values:

This function returns SUCCESSFUL when the indicated word was written correctly, or ERROR when there is a problem with the parameters.

5.4.2.10 sysWriteConfigDword

This function allows the caller to write individual longs (32 bits) to the configuration space of a specific device. The Register Number parameter must be a multiple of four (i.e., bits 0 and 1 must be set to "0").

Calling convention:

```
int sysWriteConfigDword (
        int     bus_number,
        int     device_number,
        int     function_number,
        int     register_number,            /* 0,4,8,...,252 */
        UINT32      *data
        );
```

Return values:

This function returns SUCCESSFUL when the indicated long was written correctly, or ERROR when there is a problem with the parameters.

5.4.2.11 sysGetIrqRoutingOptions

The PCI Interrupt routing fabric on the IQ80960RM/RN platform is not reconfigurable (fixed mapping relationships); therefore, this function is not supported.

Calling convention:

```
int sysGetIrqRoutingOptions (
        PCI_IRQ_ROUTING_TABLE *table
        );
```

Return values:

This function always returns FUNC_NOT_SUPPORTED.

5.4.2.12 sysSetPCIIrq

The PCI Interrupt routing fabric on the IQ80960RM/RN platform is not reconfigurable (fixed mapping relationships); therefore, this function is not supported.

Calling convention:

```
int sysSetPCIIrq (
        int     int_pin,
        int     irq_num,
        int     bus_dev
        );
```

Return values:

This function always returns FUNC_NOT_SUPPORTED.

5.4.3 Additional MON960 Commands

The following commands have been added to the UI interface of MON960 to support the IQ80960RM/RN platform.

5.4.3.1 print_pci Utility

A print_pci command to MON960 is accessed through the MON960 command prompt. This command displays the contents of the PCI configuration space on a selected adapter on the secondary PCI interface or on the i960 RM/RN I/O processor itself. For more information on the meaning of the fields in PCI configuration space, refer to the *PCI Local Bus Specification* Revision 2.1. The syntax of this command is:

pp <bus number> <device number> <function number>

5.5 Diagnostics / Example Code

IQ80960RM/RN platform diagnostic routines serve a twofold purpose: to verify proper hardware operation and to provide example code for users who need similar functions in their applications. Diagnostic routines fall into two categories: board level diagnostics and PCI expansion module diagnostics.

5.5.1 Board Level Diagnostics

Board level diagnostics exercise all basic areas of the IQ80960RM/RN platform. Diagnostic routines include SDRAM tests, UART tests, LED tests, internal timer tests, I^2C bus tests, and primary PCI bus tests. Primary PCI bus tests exercise the primary ATU, the PCI Doorbell unit, and the PCI DMA controller. Interrupts from both local and PCI sources are generated and handled. The PCI bus tests require an external test suite running on a PC to verify complete functionality of the IQ80960RM/RN platform.

5.5.2 Secondary PCI Diagnostics

Secondary PCI diagnostics exercise the secondary PCI bus, thereby confirming hardware functionality, as well as illustrating the use of the PCI BIOS routines present in MON960.

Bill of Materials A

This appendix identifies all components on the IQ80960RN Evaluation Platform (Table A-1), and the IQ80960RM Evaluation Platform (Table A-2).

Table A-1. IQ80960RN Bill of Materials (Sheet 1 of 4)

Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
1	1	U13	IC/SM 74ALS32 SOIC-14	National Semiconductor	DM74ALS32M
2	1	U6	IC/SM 74ALS04 SOIC	National Semiconductor	DM74ALS04BM
3	1	U3	IC/SM 74ABT273 SOIC	Texas Instruments	SN74ABT273DW
4	2	U1,U2	IC/SM 74ABT573 SOIC	Texas Instruments	SN74ABT573DW
5	1	U16	IC/SM 74ALS08 SOIC	National Semiconductor	DM74ALS08M
6	1	U5	IC / SM 1488A SOIC	National Semiconductor	DS1488M
7	1	U7	IC / SM 1489A SOIC	National Semiconductor	DS1489AM
8	1	Q1	IC/SM Si9430DY SOIC-8	Siliconix	Si9430DY
9	1	U9	IC/SM LVCMOS Fanout Buffr SSOP	Motorola	MPC9140
10	1	U10	IC/SM LM339 SOIC-14	National Semiconductor	LM339M
11	1	U8	IC/SM MAX1651CSA SOIC-8	Maxim	MAX1651CSA
12	1	U14	IC/SM MAX712CSE SOIC-16	Maxim	MAX712CSE
13	1	U17	IC/SM MAX767CAP SOIC	Maxim	MAX767CAP
14	1	U15	PROCESSOR (from Intel) 80960RN	Intel	
15	1	U12	VLSI I/O UART 16C550 PLCC	Texas Instruments	TL16C550AFN
16	1	C65	CAP SM, 0.47 µF (1206) Philips	Philips	12062F474Z9BB0
17	15	C2, C3, C10, C11, C18, C19, C26, C27, C55, C58, C61, C68, C77, C83, C96	CAP SM, 0.01 µF (0805)	Kemet	C0805C103K5RAC

Table A-1. IQ80960RN Bill of Materials (Sheet 2 of 4)

Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
18	81	C1, C4, C5, C6, C7, C8, C9, C12, C13, C14, C15, C16, C17, C20, C21, C22, C23, C24, C25, C28, C29, C30, C31, C32, C33, C34, C35, C36, C37, C38, C39, C40, C41, C42, C43, C44, C45, C46, C48, C49, C50, C51, C53, C59, C62, C66, C67, C69, C70, C71, C73, C79, C80, C81, C85, C86, C87, C94, C95, C97, C98, C99, C100, C101, C102, C103, C104, C105, C106, C107, C108, C109, C111, C112, C113, C115, C116, C114, C117, C120, C121	CAP SM, 0.1 µF (0805)	Philips	08052R104K8BB2
19	1	C110	CAP SM, 18 pF (0805)	Kemet	C0805C180J5GAC
20	2	R27, R28	R/SM 1/10 W 5% 1 ohm (0805)	Dale	CRCW0805100JT
21	1	R60	R/SM 1/10 W 5% 10 ohm (0805)	Dale	CRCW08051000JT
22	3	R25, R61, R62	R/SM 1/10 W 5% 1 Kohm (0805)	Dale	CRCW08051001FRT
23	4	R35, R39, R58, R59	R/SM 1/10 W 5% 10 Kohm (0805)	Dale	CRCW08051002FRT
24	2	R24, R32	R/SM 1/10 W 5% 100 Kohm (0805)	Dale	CRCW08051003FRT
25	1	R20	R/SM 1/10 W 1% 150 ohm (0805)	Dale	CRCW08051500FRT
26	3	R14, R41, R42	R/SM 1/10 W 5% 1.5 Kohm (0805)	Dale	CRCW0805152JT
27	1	R18	R/SM 1/10 W 5% 1.6 Kohm (0805)	Dale	CRCW0805162JT
28	2	R50, R51	R/SM 1/10 W 5% 22 ohm (0805)	Dale	CRCW0805220JT
29	1	R34	R/SM 1/10 W 5% 22 Kohm (0805)	Dale	CRCW0805223JT
30	1	R37	R/SM 1/10 W 5% 24 ohm (0805)	Dale	CRCW0805240JT

Table A-1. IQ80960RN Bill of Materials (Sheet 3 of 4)

Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
31	1	R47	R/SM 1/10 W 5% 2.4 Kohm (0805)	Dale	CRCW0805242JT
32	2	R2, R57	R/SM 1/10 W 5% 2.7 Kohm (0805)	Dale	CRCW0805272JT
33	1	R19	R/SM 1/10 W 5% 330 ohm (0805)	Dale	CRCW0805331JT
34	1	R29	R/SM 1/10 W 5% 36 ohm (0805)	Dale	CRCW0805360JT
35	1	R17	R/SM 1/10 W 5% 470 ohm (0805)	Dale	CRCW 0805 471JT
36	2	R48, R49	R/SM 1/10 W 1% 4.7 Kohm (0805)	Dale	CRCW08054701FRT
37	1	R53	R/SM 1/10 W 5% 47 Kohm (0805)	Dale	CRCW0805473JT
38	1	R26	R/SM 1/10 W 5% 68 Kohm (0805)	Dale	CRCW0805683JT
39	4	R30, R43, R54, R56	R/SM 1/8 W 5% 10 ohm chip 1206	Dale	CRCW1206100FT
40	5	J8, J9, J10, J11, J12	CONN SM/TH Mictor 43P Recptcl	AMP	767054-1
41	4	J1, J2, J3, J4	CONN PCI 64BIT 5 V/PCB ThruHole	AMP	145166-4
42	1	J5	CONN DIMM 168P/RAng/Socket/TH	Molex	73790-0059
43	1	J7	CONN TJ6 PCB 6/6 LP thru hole	KYCON	GM-N-66
44	1	J13	CONN/FAN ASSY/Socket/ThruHole	AMP	173981-03
45	1	J6	CONN Hdr 16 pin/w shell, pcb	AMP	103308-3
46	4	Z1, Z2, Z3, Z4	Jumper JUMP2X1	Molex	22-54-1402
47	1	L1	Inductor/SM 47µH 20%	Coilcraft	D03340P-473
48	1	L2	Inductor/SM 3.3 µH 20%	Coilcraft	D03316P-332
49	1	S1	Switch/SM DIP4 Mors# DHS-4S	Mors	DHS-4S
50	1	U4	OSC 1.8432 MHz 1/2 - Thru hole	Kyocera	KH0HC1CSE 1.843
51	1	U18	Clock Chip CY7B9910-7SC	Cypress	CY7B9910-7SC
52	1	CR5	LED Green	Hewlett Packard	HLMP-3507$010
53	1	CR3	LED-Red	Hewlett Packard	HLMP3301$010
54	1	CR4	LED Green LP	Hewlett Packard	HLMP4740#010
55	2	CR1, CR2	LED-Red-Small Group	Dialight	555-4001
56	2	Q2, Q3	Transistor/SM N-Channel	Harris	RFD16N05LSM
57	1	Q4	Transistor 2N6109 (Thru Hole)	Motorola	2N6109
58	1	U19	SOCKET PLCC20 LP Surface Mount	AMP	822269-1
60	8	BT1, BT2, BT3, BT4, BT5, BT6, BT7, BT8	Battery Clips/PC/Snap-In/AA	Keystone	#92
61	1	U19	PALLV16V8Z-20JI	AMD	PALLV16V8Z-20JI
62	1	U11	MEM Flash E28F016S5-090 TSOP	Intel	E28F016S5-090

Table A-1. IQ80960RN Bill of Materials (Sheet 4 of 4)

Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
63	8	BT1, BT2, BT3, BT4, BT5, BT6, BT7, BT8	Battery AA NiCd @ 600 mA/Hour	SAFT	NIC-AA-600-SAFT
64	1	U15	HeatSink/Fan Assy 80960RM/RN	Panasonic	UDQFNBEOIF
65	1	C84	CAP SM, 0.22 µF (1206)	Philips	12062E224M9BB2
66	3	C60, C75, C78	CAP TANT SM 220 µF, 10 V (7343)	AVX	TPSE227K010R010
67	4	C89, C90, C91, C93	CAP TANT SM 47 µF, 16 V (7343)	AVX	TPSD476K016R015
68	1	C63	CAP TANT SM 33 µF, 10 V (7343)	Sprague	293D336X9016D2T
69	4	C57, C76, C88, C92	CAP TANT SM 4.7 µF, 35 V (7343)	Sprague	293D475X9035D2T
70	1	C47	CAP TANT SM 22 µF, 20 V (7343)	Sprague	293D226X9020D2T
71	1	C74	CAP TANT SM 1 µF, 16 V (3216)	Sprague	293D105X0016A2T
72	2	C52, C54	CAP TANT SM 10 µF, 25/35 V	Sprague	293D1060025D2T
73	1	C56	CAP TANT SM 100 µF 10 V (7343)	AVX	TPSD107K010R0100
74	1	C64	CAP TANT SM 330 µF 6.3 V (7343	AVX	TPSE337K063R0100
75	3	C82, C118, C119	CAP SM, 0.047 µF (0805)	Kemet	C0805C473K5RAC
76	1	R46	Res/SM 1 W 1% 0.012 ohm (2512)	Dale	WSL-2512-R012
77	1	R21	Res/SM 1 W 1% 0.05 ohm (2512)	Dale	WSL-2512-R050
78	1	R52	Resistor/SM 1/2 W 5% 100 ohm	Beckmen	BCR 1/2 101 JT
79	16	R1, R3, R4, R5, R6, R7, R8, R9, R10, R11, R12, R33, R36, R38, R44, R45,	Resistor Pk SM RNC4R8P 2.7 Kohm	CTS	742083272JTR
80	2	R40, R55	Resistor Pk SM RNC4R8P 22 ohm	CTS	742083220JTR
81	2	R15, R16	Resistor Pk SM RNC4R8P 470 ohm	CTS	742083471JTR
82	1	R13	Resistor Pk SM RNC4R8P 1.5 Kohm	CTS	742083152JTR
83	2	R22, R23	Resistor Pk SM RNC4R8P 30 ohm	CTS	742083300JTR
84	1	CR9	Diode CMPSH3 Surface Mount	Central Semiconductor	CMPSH3
85	2	CR6, CR7	Diode SM / MBRS340T3	Motorola	MBRS340T3
86	1	CR8	Diode/SM 1N4001 (CMR1-02)	Central Semiconductor	CMR1-02
87	1	J5	SDRAM, DIMM, ECC, 2Mx72, 16 MB	Unigen	UG52S7408GSG
88	2	U20, U21	IC/SM TL7702BCD	Texas Instruments	TL7702BCD

Table A-2. IQ80960RM Bill of Materials (Sheet 1 of 5)

Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
1	1	U13	IC/SM 74ALS32 SOIC-14	National Semiconductor	DM74ALS32M
2	1	U6	IC/SM 74ALS04 SOIC	National Semiconductor	DM74ALS04BM
3	1	U3	IC/SM 74ABT273 SOIC	Texas Instruments	SN74ABT273DW
4	2	U1, U2	IC/SM 74ABT573 SOIC	Texas Instruments	SN74ABT573DW
5	1	U16	IC/SM 74ALS08 SOIC	National Semiconductor	DM74ALS08M
6	1	U5	IC / SM 1488A SOIC	National Semiconductor	DS1488M
7	1	U7	IC / SM 1489A SOIC	National Semiconductor	DS1489AM
8	1	Q1	IC/SM Si9430DY SOIC-8	Siliconix	Si9430DY
9	1	U9	IC/SM LVCMOS Fanout Buffr SSOP	Motorola	MPC9140
10	1	U10	IC/SM LM339 SOIC-14	National Semiconductor	LM339M
11	1	U8	IC/SM MAX1651CSA SOIC-8	Maxim	MAX1651CSA
12	1	U14	IC/SM MAX712CSE SOIC-16	Maxim	MAX712CSE
13	1	U17	IC/SM MAX767CAP SOIC	Maxim	MAX767CAP
14	1	U15	PROCESSOR (frm Intel) i960RM	Intel	
15	1	U12	VLSI I/O UART 16C550 PLCC	Texas Instruments	TL16C550AFN
16	1	C65	CAP SM, 0.47 µF (1206) Philips	Philips	12062F474Z9BB0
17	15	C2, C3, C10, C11, C18, C19, C26, C27, C55, C58, C61, C68, C77, C83, C96	CAP SM, 0.01 µF (0805)	Kemet	C0805C103K5RAC

Table A-2. IQ80960RM Bill of Materials (Sheet 2 of 5)

Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
18	81	C1, C4, C5, C6, C7, C8, C9, C12, C13, C14, C15, C16, C17, C20, C21, C22, C23, C24, C25, C28, C29, C30, C31, C32, C33, C34, C35, C36, C37, C38, C39, C40, C41, C42, C43, C44, C45, C46, C48, C49, C50, C51, C53, C59, C62, C66, C67, C69, C70, C71, C73, C79, C80, C81, C85, C86, C87, C94, C95, C97, C98, C99, C100, C101, C102, C103, C104, C105, C106, C107, C108, C109, C111, C112, C113, C114, C115, C116, C117, C120, C121	CAP SM, 0.1 µF (0805)	Philips	08052R104K8BB2
19	1	C110	CAP SM, 18 pF(0805)	Kemet	C0805C180J5GAC
20	2	R27, R28	R/SM 1/10 W 5% 1 ohm (0805)	Dale	CRCW0805100JT
21	1	R60	R/SM 1/10 W 5% 10 ohm (0805)	Dale	CRCW08051000JT
22	3	R25, R61, R62	R/SM 1/10 W 5% 1 Kohm (0805)	Dale	CRCW08051001FRT
23	12	R5, R6, R7 R8, R9, R10, R11, R12, R35, R39, R58, R59	R/SM 1/10 W 5% 10 Kohm (0805)	Dale	CRCW08051002FRT
24	2	R24, R32	R/SM 1/10 W 5% 100 Kohm (0805)	Dale	CRCW08051003FRT
25	1	R20	R/SM 1/10 W 1% 150 ohm (0805)	Dale	CRCW08051500FRT
26	3	R14, R41, R42	R/SM 1/10 W 5% 1.5 Kohm (0805)	Dale	CRCW0805152JT
27	1	R18	R/SM 1/10 W 5% 1.6 Kohm (0805)	Dale	CRCW0805162JT

Table A-2. IQ80960RM Bill of Materials (Sheet 3 of 5)

Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
28	2	R50, R51	R/SM 1/10 W 5% 22 ohm (0805)	Dale	CRCW0805220JT
29	1	R34	R/SM 1/10 W 5% 22 Kohm (0805)	Dale	CRCW0805223JT
30	1	R37	R/SM 1/10 W 5% 24 ohm (0805)	Dale	CRCW0805240JT
31	1	R47	R/SM 1/10 W 5% 2.4 Kohm (0805)	Dale	CRCW0805242JT
32	1	R57	R/SM 1/10 W 5% 2.7 Kohm (0805)	Dale	CRCW0805272JT
33	1	R19	R/SM 1/10 W 5% 330 ohm (0805)	Dale	CRCW0805331JT
34	1	R29	R/SM 1/10 W 5% 36 ohm (0805)	Dale	CRCW0805360JT
35	1	R17	R/SM 1/10 W 5% 470 ohm (0805)	Dale	CRCW 0805 471JT
36	2	R48, R49	R/SM 1/10 W 1% 4.7 Kohm (0805)	Dale	CRCW08054701FRT
37	1	R53	R/SM 1/10 W 5% 47 Kohm (0805)	Dale	CRCW0805473JT
38	1	R26	R/SM 1/10 W 5% 68 Kohm (0805)	Dale	CRCW0805683JT
39	4	R30, R43, R54, R56	R/SM 1/8 W 5% 10 ohm chip 1206	Dale	CRCW1206100FT
40	5	J8, J9, J10, J11, J12	CONN SM/TH Mictor 43P Recptcl	AMP	767054-1
41	4	J1, J2, J3, J4	CONN PCI Slot 5V/PCB ThruHole	AMP	145154-4
42	1	J5	CONN DIMM 168P/RAng/Socket/TH	Molex	73790-0059
43	1	J7	CONN TJ6 PCB 6/6 LP thru hole	KYCON	GM-N-66
44	1	J13	CONN/FAN ASSY/Socket/ThruHole	AMP	173981-03
45	1	J6	CONN Hdr 16 pin/w shell, pcb	AMP	103308-3
46	4	Z1, Z2, Z3, Z4	Jumper JUMP2X1	Molex	22-54-1402
47	1	L1	Inductor/SM 47 µH 20%	Coilcraft	D03340P-473
48	1	L2	Inductor/SM 3.3 µH 20%	Coilcraft	D03316P-332
49	1	S1	Switch/SM DIP4 Mors# DHS-4S	Mors	DHS-4S
50	1	U4	OSC 1.8432 MHz 1/2 - Thru hole	Kyocera	KH0HC1CSE 1.843
51	1	U18	Clock Chip CY7B9910-7SC	Cypress	CY7B9910-7SC
52	1	CR5	LED Green	Hewlett Packard	HLMP-3507$010
53	1	CR3	LED-Red	Hewlett Packard	HLMP3301$010
54	1	CR4	LED Green LP	Hewlett Packard	HLMP4740#010
55	2	CR1, CR2	LED-Red-Small Group	Dialight	555-4001
56	2	Q2, Q3	Transistor/SM N-Channel	Harris	RFD16N05LSM
57	1	Q4	Transistor 2N6109 (Thru Hole)	Motorola	2N6109
58	1	U19	SOCKET PLCC20 LP Surface Mount	AMP	822269-1
60	1	U11	SOCKET / SM / TSOP / 40 pin	Meritec	980020-40-02

Table A-2. IQ80960RM Bill of Materials (Sheet 4 of 5)

Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
61	8	BT1, BT2, BT3, BT4, BT5, BT6, BT7, BT8	Battery Clips/PC/Snap-In/AA	Keystone	#92
62	1	U19	PALLV16V8Z-20JI	AMD	PALLV16V8Z-20JI
63	1	U11	MEM Flash E28F016S5-090 TSOP	Intel	E28F016S5-090
64	8	BT1, BT2, BT3, BT4, BT5, BT6, BT7, BT8	Battery AA NiCd @ 600 mA/Hour	SAFT	NIC-AA-600-SAFT
65	1	U15	HeatSink/Fan Assy 80960RN/RM	Panasonic	UDQFNBEOIF
66	3	C84	CAP SM, 0.22 µF (1206)	Philips	12062E224M9BB2
67	3	C60, C75, C78	CAP TANT SM 220 µF, 10 V (7343)	AVX	TPSE227K010R010
68	4	C89, C90, C91, C93	CAP TANT SM 47 µF, 16 V (7343)	AVX	TPSD476K016R015
69	1	C63	CAP TANT SM 33 µF, 10 V (7343)	Sprague	293D336X9016D2T
70	4	C57, C76, C88, C92	CAP TANT SM 4.7 µF, 35 V (7343)	Sprague	293D475X9035D2T
71	1	C47	CAP TANT SM 22 µF, 20 V (7343)	Sprague	293D226X9020D2T
72	1	C74	CAP TANT SM 1 µF, 16 V (3216)	Sprague	293D105X0016A2T
73	2	C52, C54	CAP TANT SM 10 µF, 25/35 V	Sprague	293D1060025D2T
74	1	C56	CAP TANT SM 100 µF 10 V (7343)	AVX	TPSD107K010R0100
75	1	C64	CAP TANT SM 330 µF 6.3 V (7343)	AVX	TPSE337K063R0100
76	3	C82, C118, C119	CAP SM, 0.047 µF (0805)	Kemet	C0805C473K5RAC
77	1	R46	Res/SM 1 W 1% 0.012 ohm (2512)	Dale	WSL-2512-R012
78	1	R21	Res/SM 1 W 1% 0.05 ohm (2512)	Dale	WSL-2512-R050
79	1	R52	Resistor/SM 1/2 W 5% 100 ohm	Beckmen	BCR 1/2 101 JT
80	7	R1, R31, R33, R36, R38, R44, R45	Resistor Pk SM RNC4R8P 2.7 Kohm	CTS	742083272JTR
81	2	R40, R55	Resistor Pk SM RNC4R8P 22 ohm	CTS	742083220JTR
82	2	R15, R16	Resistor Pk SM RNC4R8P 470 ohm	CTS	742083471JTR
83	1	R13	Resistor Pk SM RNC4R8P 1.5 Kohm	CTS	742083152JTR
84	2	R22, R23	Resistor Pk SM RNC4R8P 30 ohm	CTS	742083300JTR
85	1	CR9	Diode CMPSH3 Surface Mount	Central Semiconductor	CMPSH3
86	2	CR6, CR7	Diode SM / MBRS340T3	Motorola	MBRS340T3
87	1	CR8	Diode/SM 1N4001 (CMR1-02)	Central Semiconductor	CMR1-02
88	1	J5	SDRAM, DIMM, ECC, 2Mx72, 16 MB	Unigen	UG52S7408GSG

Table A-2. IQ80960RM Bill of Materials (Sheet 5 of 5)

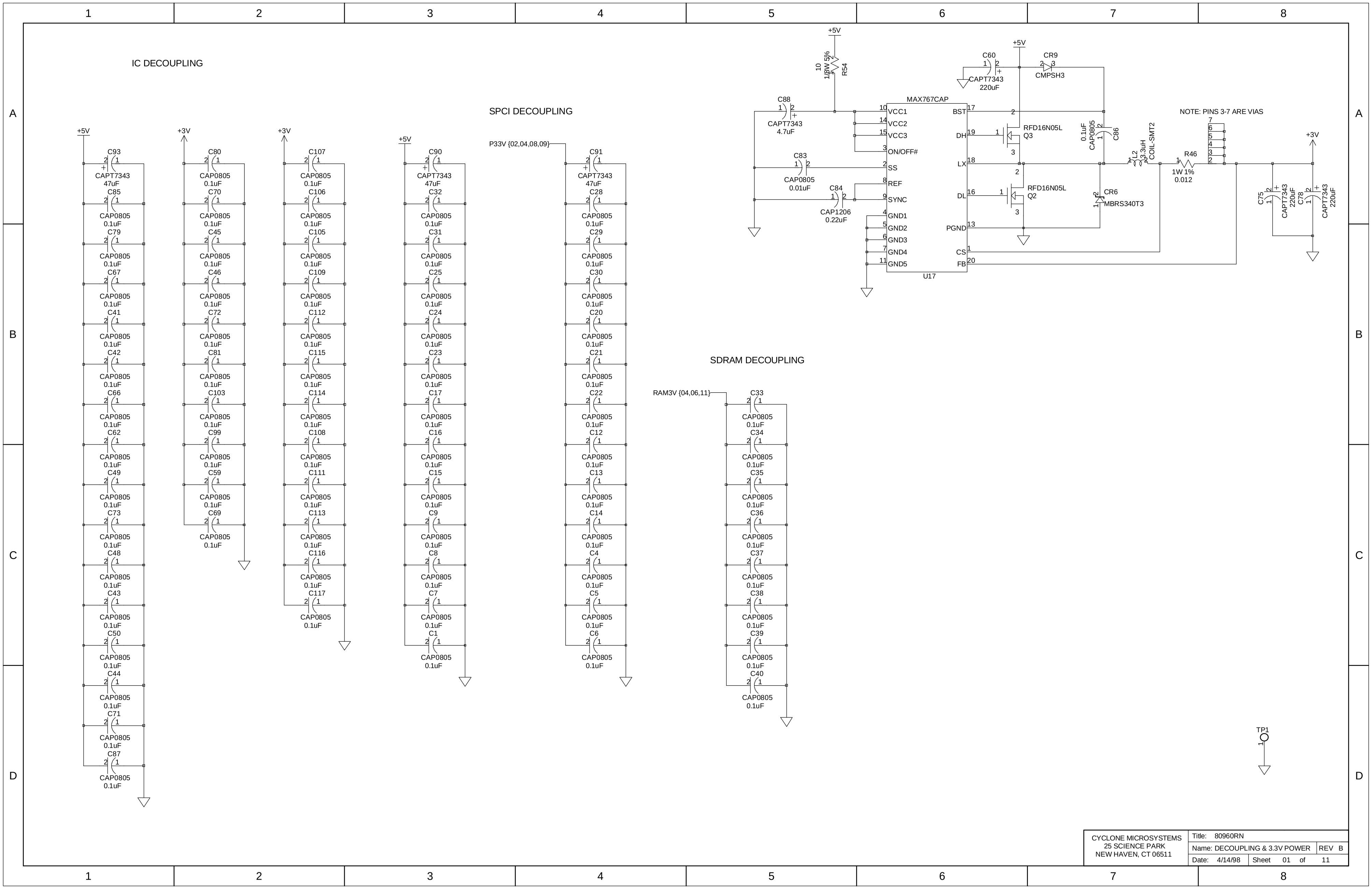
Item	Qty	Location	Part Description	Manufacturer	Manufacturer Part #
89	2	U20, U21	IC/SM TL7702BCD	Texas Instruments	TL7702BCD

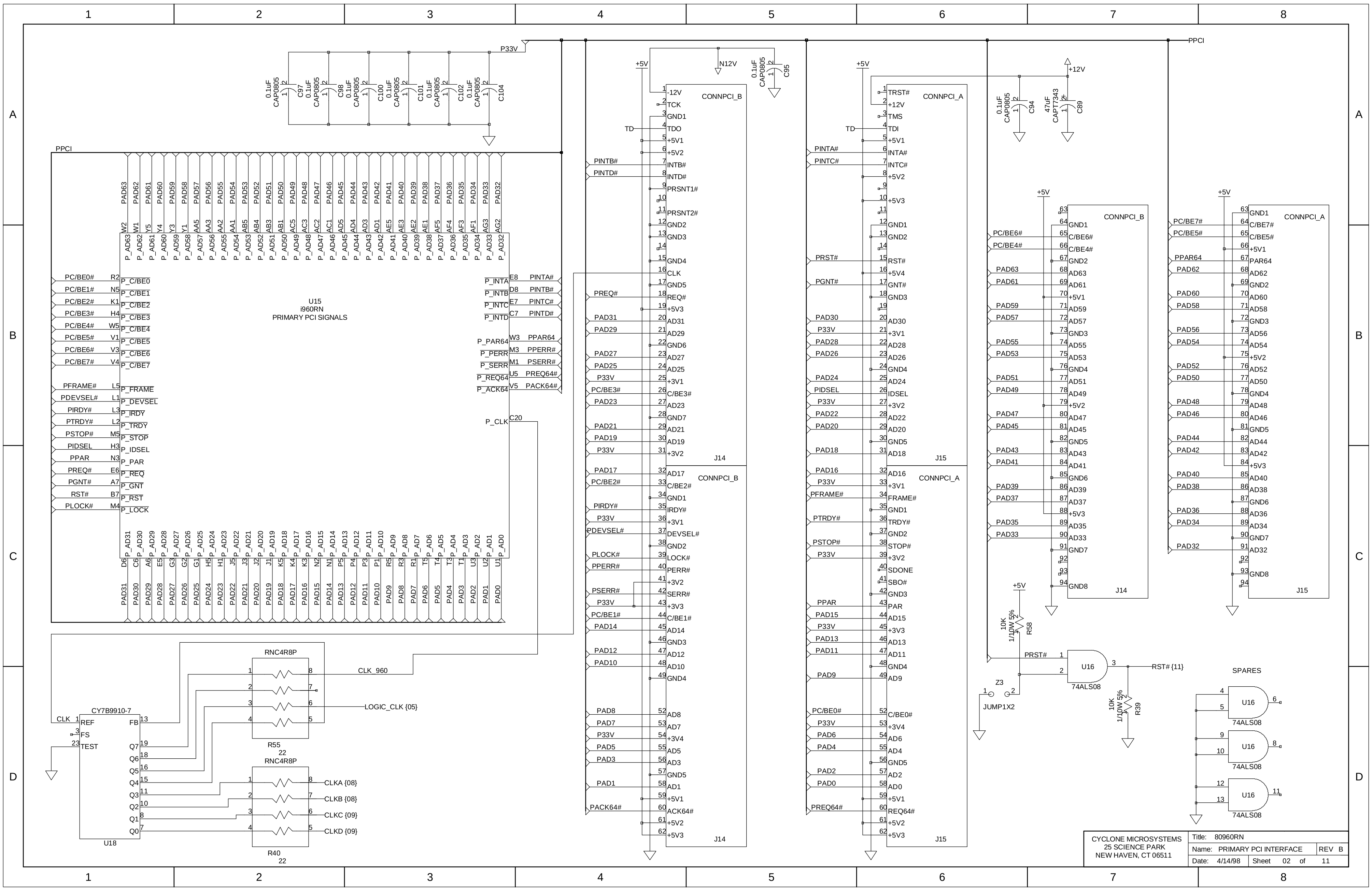
Schematics B

This appendix includes schematics for the IQ80960RN (Table B-1) and IQ80960RM (Table B-2).

Table B-1. IQ80960RN Schematics List

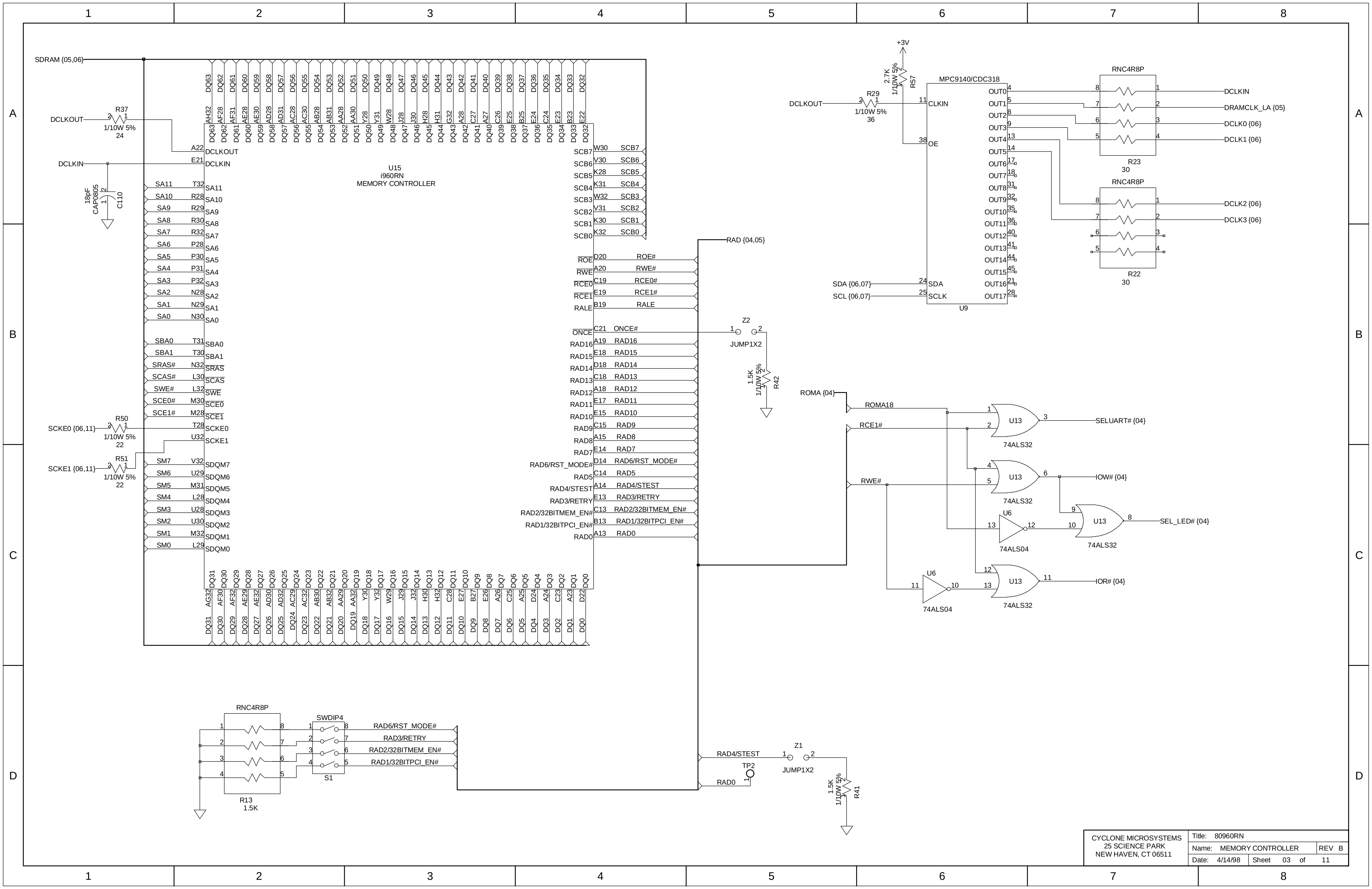
Page	Schematic Title
B-2	Decoupling and 3.3V Power
B-3	Primary PCI Interface
B-4	Memory Controller
B-5	Flash ROM, UART, & LEDs
B-6	Logic Analyzer I/F
B-7	SDRAM 168-Pin DIMM
B-8	Secondary PCI/960 Core
B-9	Secondary PCI Bus 1/2
B-10	Secondary PCI Bus 3/4
B-11	SPCI Pull-ups
B-12	Battery/Monitor

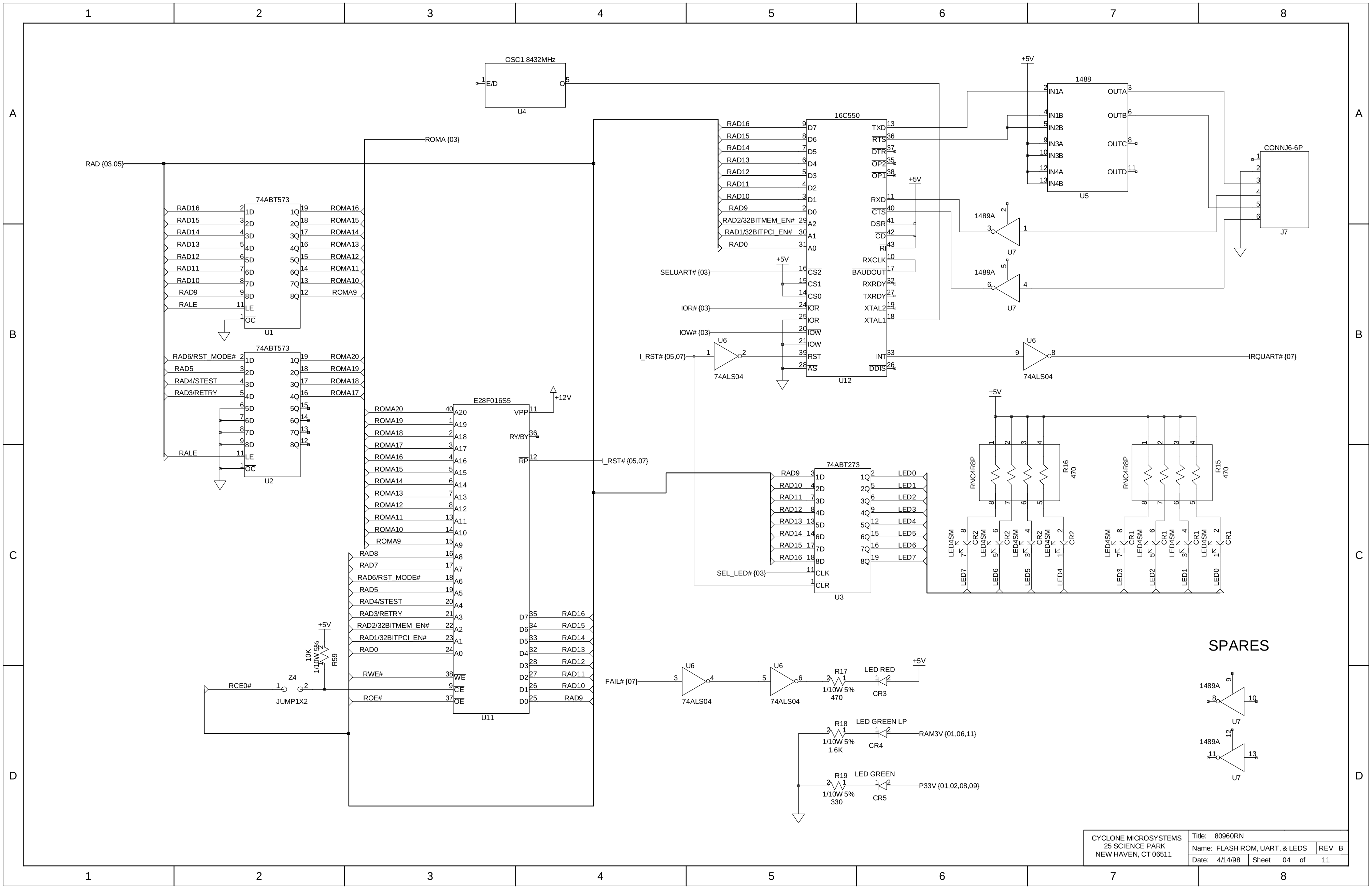

IC DECOUPLING
SPCI DECOUPLING
SDRAM DECOUPLING
+5V
+3V
P33V (02,04,08,09)
RAM3V (04,06,11)
C93 CAPT7343 47uF
C90 CAPT7343 47uF
C91 CAPT7343 47uF
CAP0805 0.1uF
MAX767CAP
U17
VCC1
VCC2
VCC3
ON/OFF#
SS
REF
SYNC
GND1
GND2
GND3
GND4
GND5
BST
DH
LX
DL
PGND
CS
FB
C88 CAPT7343 4.7uF
R54 10 1/8W 5%
C83 CAP0805 0.01uF
C84 CAP1206 0.22uF
C60 CAPT7343 220uF
CR9 CMPSH3
Q3 RFD16N05L
Q2 RFD16N05L
C86 CAP0805 0.1uF
CR6 MBRS340T3
L2 3.3uH COIL-SMT2
R46 1W 1% 0.012
NOTE: PINS 3-7 ARE VIAS
C75 CAPT7343 220uF
C78 CAPT7343 220uF
TP1
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RN
Name: DECOUPLING & 3.3V POWER
REV B
Date: 4/14/98
Sheet 01 of 11

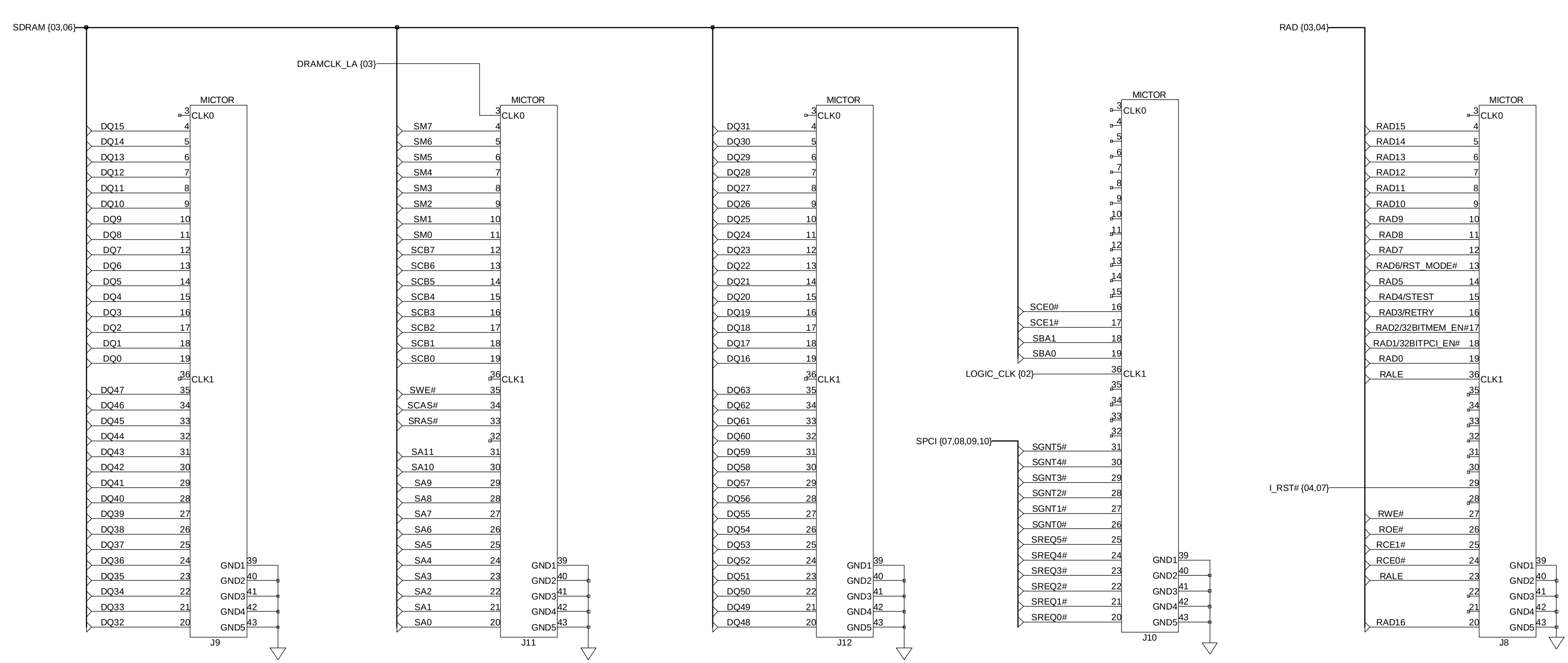


CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511

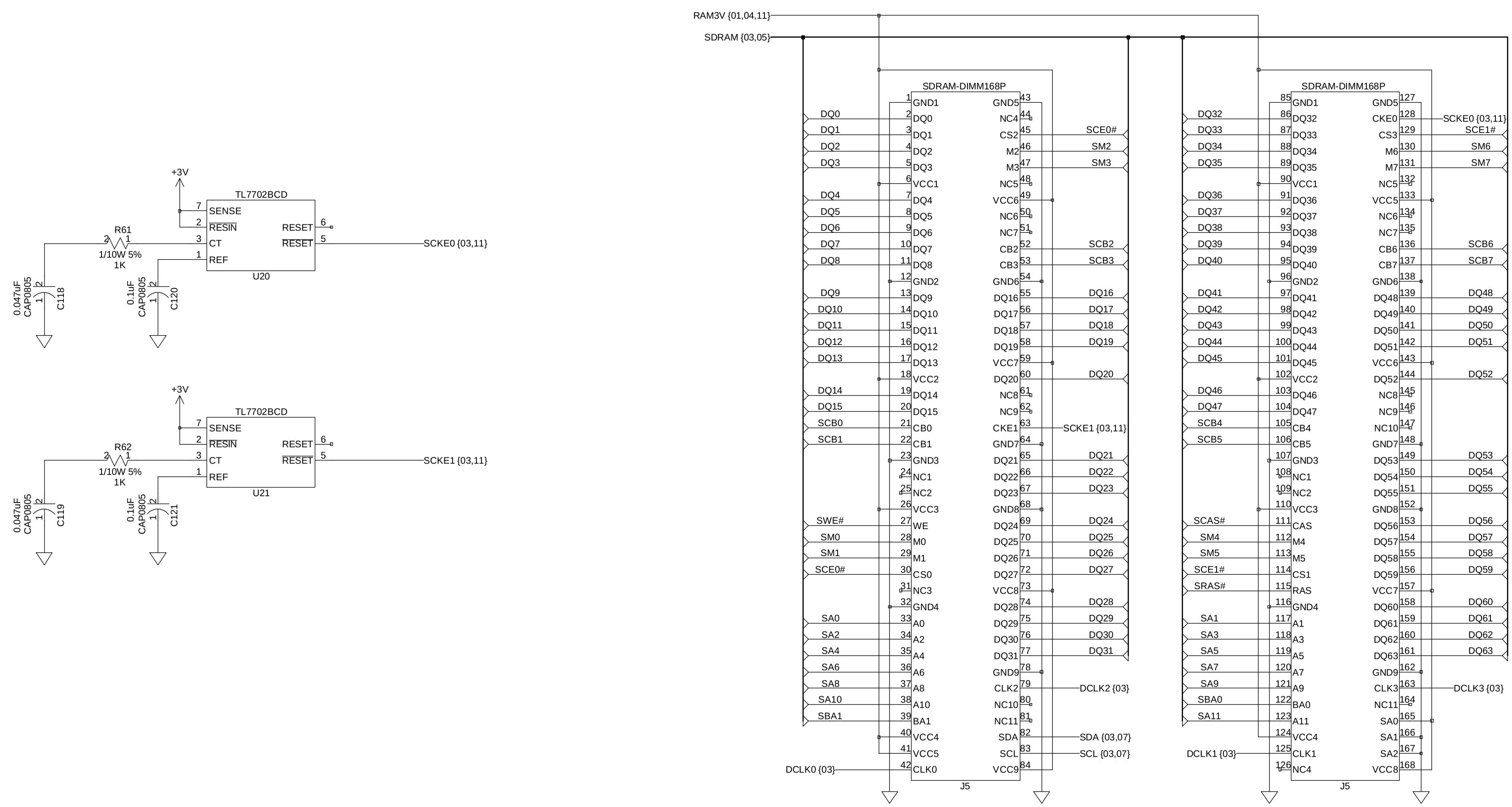
Title: 80960RN		
Name: PRIMARY PCI INTERFACE		REV B
Date: 4/14/98	Sheet 02 of 11	


U15
i960RN
MEMORY CONTROLLER
MPC9140/CDC318
U9
RNC4R8P
R23
R22
R13
SWDIP4
S1
JUMP1X2
Z1
Z2
TP2
U13
U6
74ALS32
74ALS04
SDRAM {05,06}
DCLKOUT
DCLKIN
DRAMCLK_LA {05}
DCLK0 {06}
DCLK1 {06}
DCLK2 {06}
DCLK3 {06}
SDA {06,07}
SCL {06,07}
RAD {04,05}
ROMA {04}
SELUART# {04}
IOW# {04}
SEL_LED# {04}
IOR# {04}
SCKE0 {06,11}
SCKE1 {06,11}
+3V
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RN
Name: MEMORY CONTROLLER
REV B
Date: 4/14/98
Sheet 03 of 11


SPARES
16C550
U12
74ABT573
U1
U2
E28F016S5
U11
74ABT273
U3
1488
U5
1489A
U7
74ALS04
U6
OSC1.8432MHz
U4
CONNJ6-6P
J7
RNC4R8P
R15
R16
470
LED4SM
CR1
CR2
R17
1/10W 5%
470
LED RED
CR3
R18
1.6K
LED GREEN LP
CR4
R19
330
LED GREEN
CR5
R59
10K
Z4
JUMP1X2
RAD {03,05}
ROMA {03}
SELUART# {03}
IOR# {03}
IOW# {03}
I_RST# {05,07}
IRQUART# {07}
SEL_LED# {03}
FAIL# {07}
RAM3V {01,06,11}
P33V {01,02,08,09}
RCE0#
+5V
+12V
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RN
Name: FLASH ROM, UART, & LEDS
REV B
Date: 4/14/98
Sheet 04 of 11

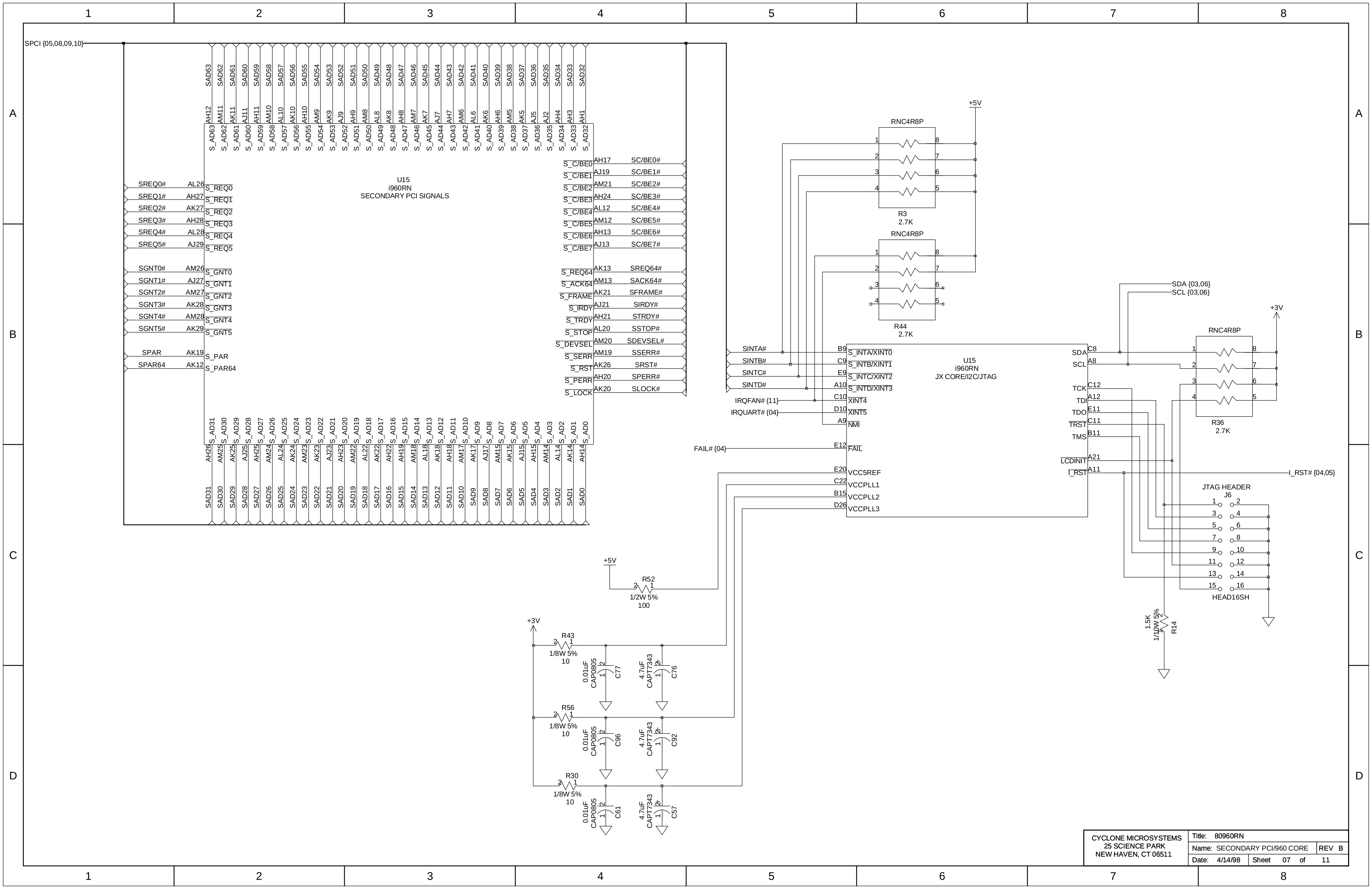


CYCLONE MICROSYSTEMS 25 SCIENCE PARK NEW HAVEN, CT 06511	Title: 80960RN	
	Name: LOGIC ANALYZER I/F	REV B
	Date: 4/14/98	Sheet 05 of 11

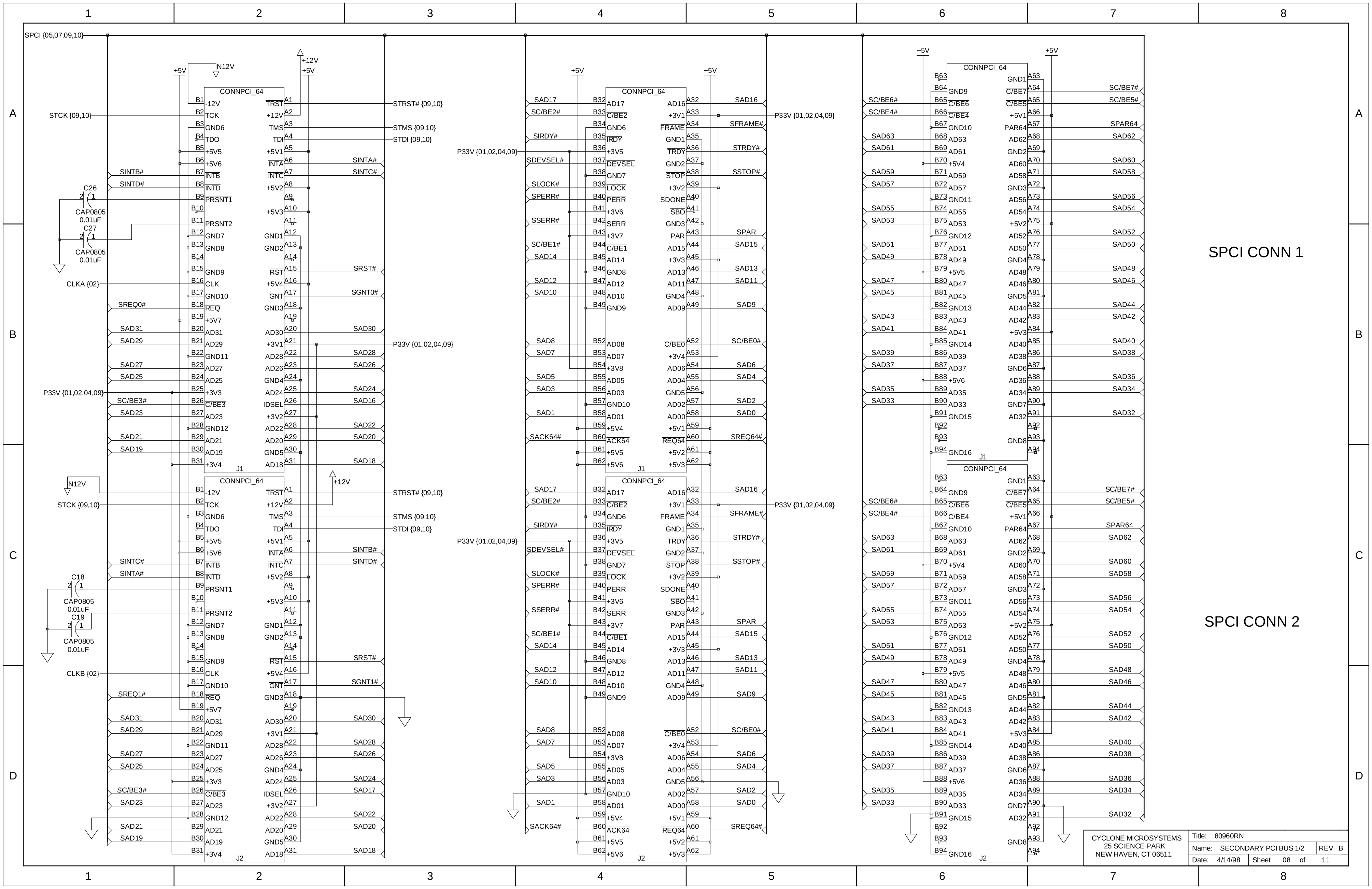


CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511

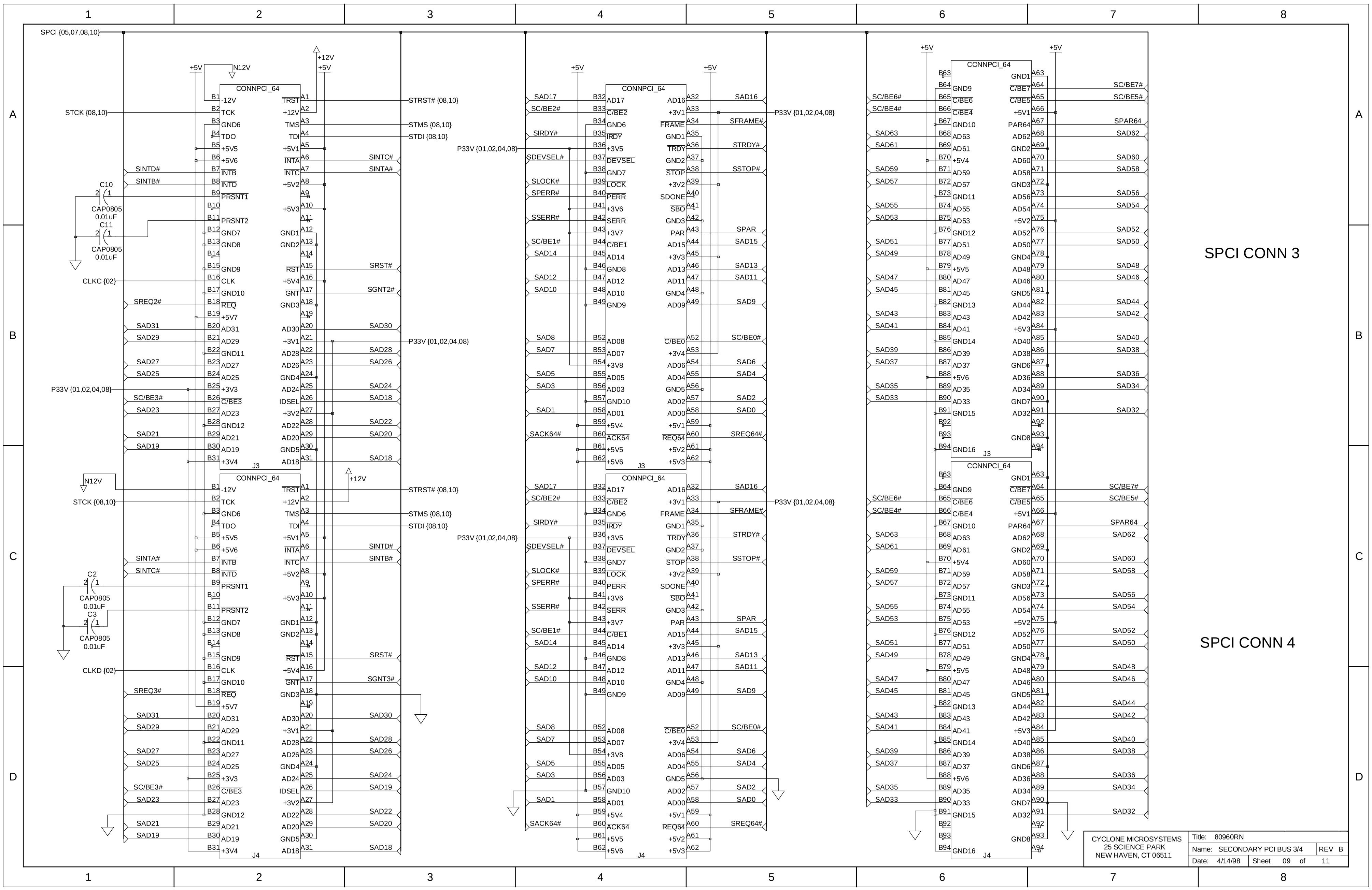
Title: 80960RN	
Name: SDRAM 168-PIN DIMM	REV C
Date: 2/4/99	Sheet 6 of 11

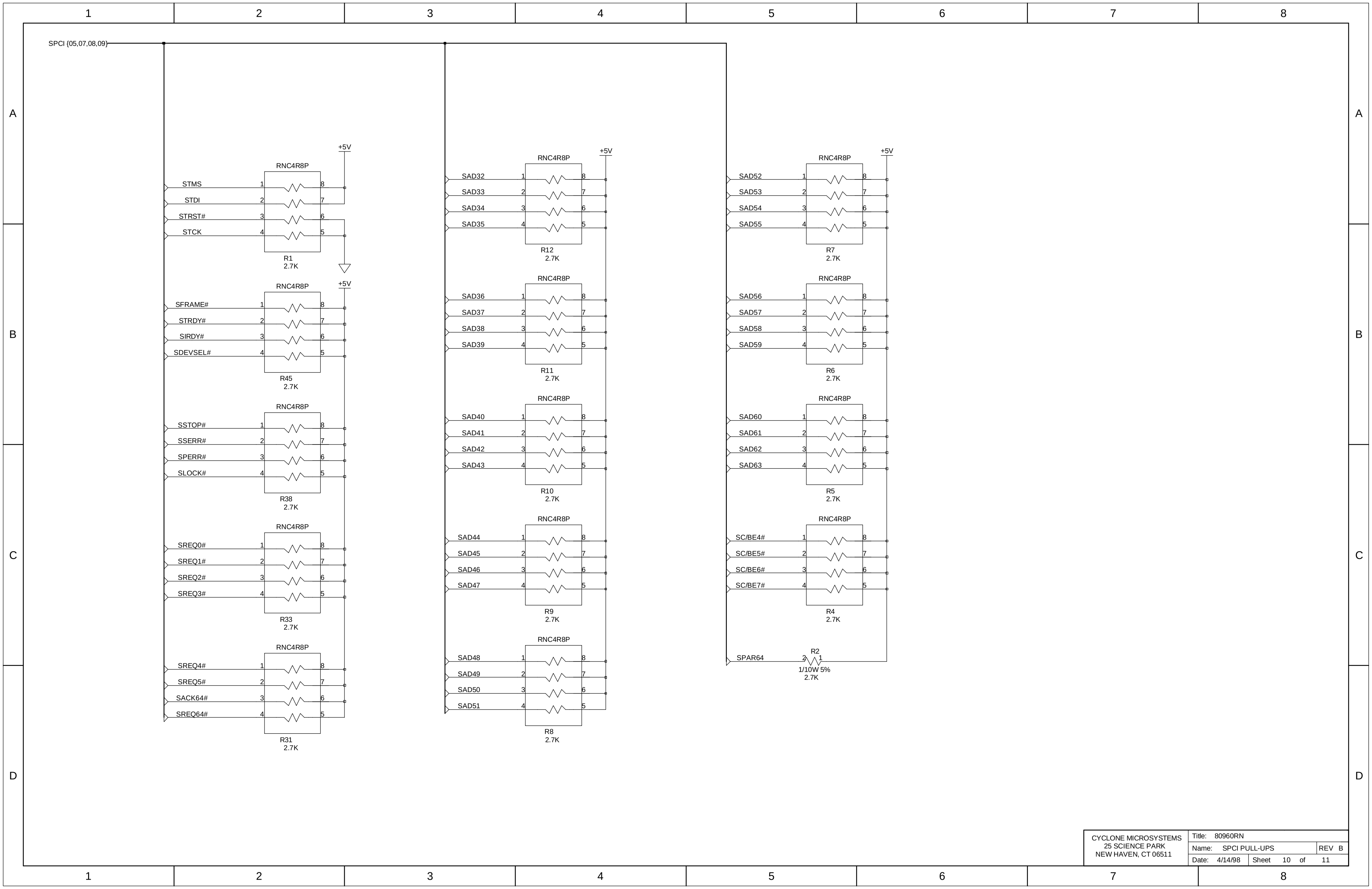


SPCI {05,08,09,10}
U15
i960RN
SECONDARY PCI SIGNALS
SAD63 SAD62 SAD61 SAD60 SAD59 SAD58 SAD57 SAD56 SAD55 SAD54 SAD53 SAD52 SAD51 SAD50 SAD49 SAD48 SAD47 SAD46 SAD45 SAD44 SAD43 SAD42 SAD41 SAD40 SAD39 SAD38 SAD37 SAD36 SAD35 SAD34 SAD33 SAD32
AH12 AM11 AK11 AJ11 AH11 AM10 AL10 AK10 AH10 AM9 AK9 AJ9 AH9 AM8 AL8 AK8 AH8 AM7 AK7 AJ7 AH7 AM6 AL6 AK6 AH6 AM5 AK5 AJ5 AJ2 AH4 AH3 AH1
S_AD63 S_AD62 S_AD61 S_AD60 S_AD59 S_AD58 S_AD57 S_AD56 S_AD55 S_AD54 S_AD53 S_AD52 S_AD51 S_AD50 S_AD49 S_AD48 S_AD47 S_AD46 S_AD45 S_AD44 S_AD43 S_AD42 S_AD41 S_AD40 S_AD39 S_AD38 S_AD37 S_AD36 S_AD35 S_AD34 S_AD33 S_AD32
SREQ0# AL26 S_REQ0
SREQ1# AH27 S_REQ1
SREQ2# AK27 S_REQ2
SREQ3# AH28 S_REQ3
SREQ4# AL28 S_REQ4
SREQ5# AJ29 S_REQ5
SGNT0# AM26 S_GNT0
SGNT1# AJ27 S_GNT1
SGNT2# AM27 S_GNT2
SGNT3# AK28 S_GNT3
SGNT4# AM28 S_GNT4
SGNT5# AK29 S_GNT5
SPAR AK19 S_PAR
SPAR64 AK12 S_PAR64
S_C/BE0 AH17 SC/BE0#
S_C/BE1 AJ19 SC/BE1#
S_C/BE2 AM21 SC/BE2#
S_C/BE3 AH24 SC/BE3#
S_C/BE4 AL12 SC/BE4#
S_C/BE5 AM12 SC/BE5#
S_C/BE6 AH13 SC/BE6#
S_C/BE7 AJ13 SC/BE7#
S_REQ64 AK13 SREQ64#
S_ACK64 AM13 SACK64#
S_FRAME AK21 SFRAME#
S_IRDY AJ21 SIRDY#
S_TRDY AH21 STRDY#
S_STOP AL20 SSTOP#
S_DEVSEL AM20 SDEVSEL#
S_SERR AM19 SSERR#
S_RST AK26 SRST#
S_PERR AH20 SPERR#
S_LOCK AK20 SLOCK#
S_AD31 S_AD30 S_AD29 S_AD28 S_AD27 S_AD26 S_AD25 S_AD24 S_AD23 S_AD22 S_AD21 S_AD20 S_AD19 S_AD18 S_AD17 S_AD16 S_AD15 S_AD14 S_AD13 S_AD12 S_AD11 S_AD10 S_AD9 S_AD8 S_AD7 S_AD6 S_AD5 S_AD4 S_AD3 S_AD2 S_AD1 S_AD0
AH26 AM25 AK25 AJ25 AH25 AM24 AL24 AK24 AM23 AK23 AJ23 AH23 AM22 AL22 AK22 AH22 AH19 AM18 AL18 AK18 AH18 AM17 AK17 AJ17 AM15 AK15 AJ15 AH15 AM14 AL14 AK14 AH14
SAD31 SAD30 SAD29 SAD28 SAD27 SAD26 SAD25 SAD24 SAD23 SAD22 SAD21 SAD20 SAD19 SAD18 SAD17 SAD16 SAD15 SAD14 SAD13 SAD12 SAD11 SAD10 SAD9 SAD8 SAD7 SAD6 SAD5 SAD4 SAD3 SAD2 SAD1 SAD0
+5V
RNC4R8P
R3
2.7K
RNC4R8P
R44
2.7K
U15
i960RN
JX CORE/I2C/JTAG
SINTA# B9 S_INTA/XINT0
SINTB# C9 S_INTB/XINT1
SINTC# E9 S_INTC/XINT2
SINTD# A10 S_INTD/XINT3
IRQFAN# {11} C10 XINT4
IRQUART# {04} D10 XINT5
A9 NMI
FAIL# {04} E12 FAIL
E20 VCC5REF
C22 VCCPLL1
B15 VCCPLL2
D26 VCCPLL3
SDA C8
SCL A8
TCK C12
TDI A12
TDO E11
TRST C11
TMS B11
LCDINIT A21
I_RST A11
SDA {03,06}
SCL {03,06}
+3V
RNC4R8P
R36
2.7K
I_RST# {04,05}
JTAG HEADER
J6
HEAD16SH
1.5K
1/10W 5%
R14
+5V
R52
1/2W 5%
100
+3V
R43
1/8W 5%
10
0.01uF
CAP0805
C77
4.7uF
CAPT7343
C76
R56
1/8W 5%
10
0.01uF
CAP0805
C96
4.7uF
CAPT7343
C92
R30
1/8W 5%
10
0.01uF
CAP0805
C61
4.7uF
CAPT7343
C57
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RN
Name: SECONDARY PCI/960 CORE
REV B
Date: 4/14/98
Sheet 07 of 11



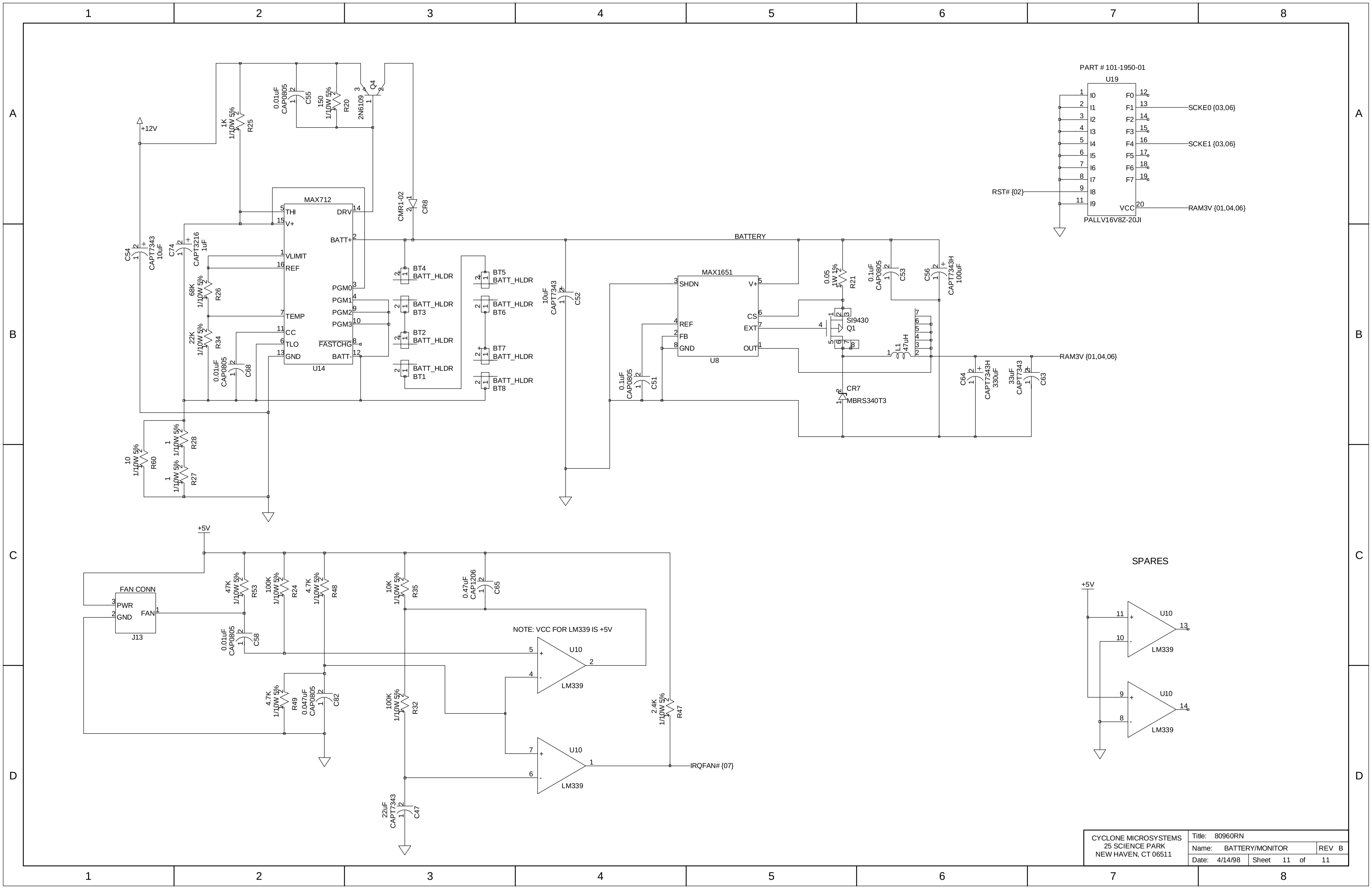
SPCI CONN 1
SPCI CONN 2
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RN
Name: SECONDARY PCI BUS 1/2
REV B
Date: 4/14/98
Sheet 08 of 11


SPCI CONN 3
SPCI CONN 4
SPCI {05,07,08,10}
STCK {08,10}
CLKC {02}
CLKD {02}
P33V {01,02,04,08}
STRST# {08,10}
STMS {08,10}
STDI {08,10}
CONNPCI_64
J3
J4
C10
C11
C2
C3
CAP0805
0.01uF
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RN
Name: SECONDARY PCI BUS 3/4
REV B
Date: 4/14/98
Sheet 09 of 11



CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511

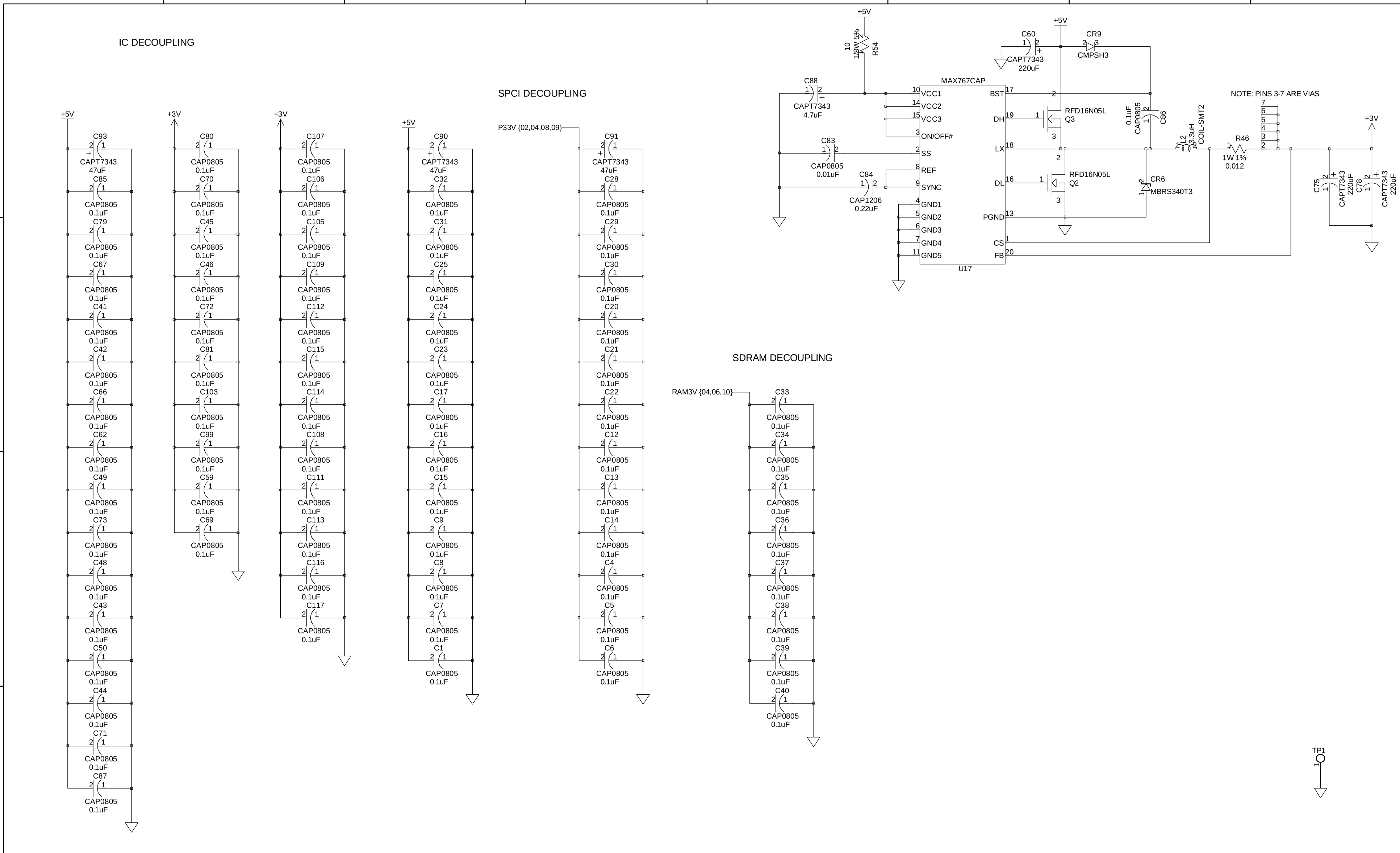
Title: 80960RN		
Name: SPCI PULL-UPS		REV B
Date: 4/14/98	Sheet 10 of 11	



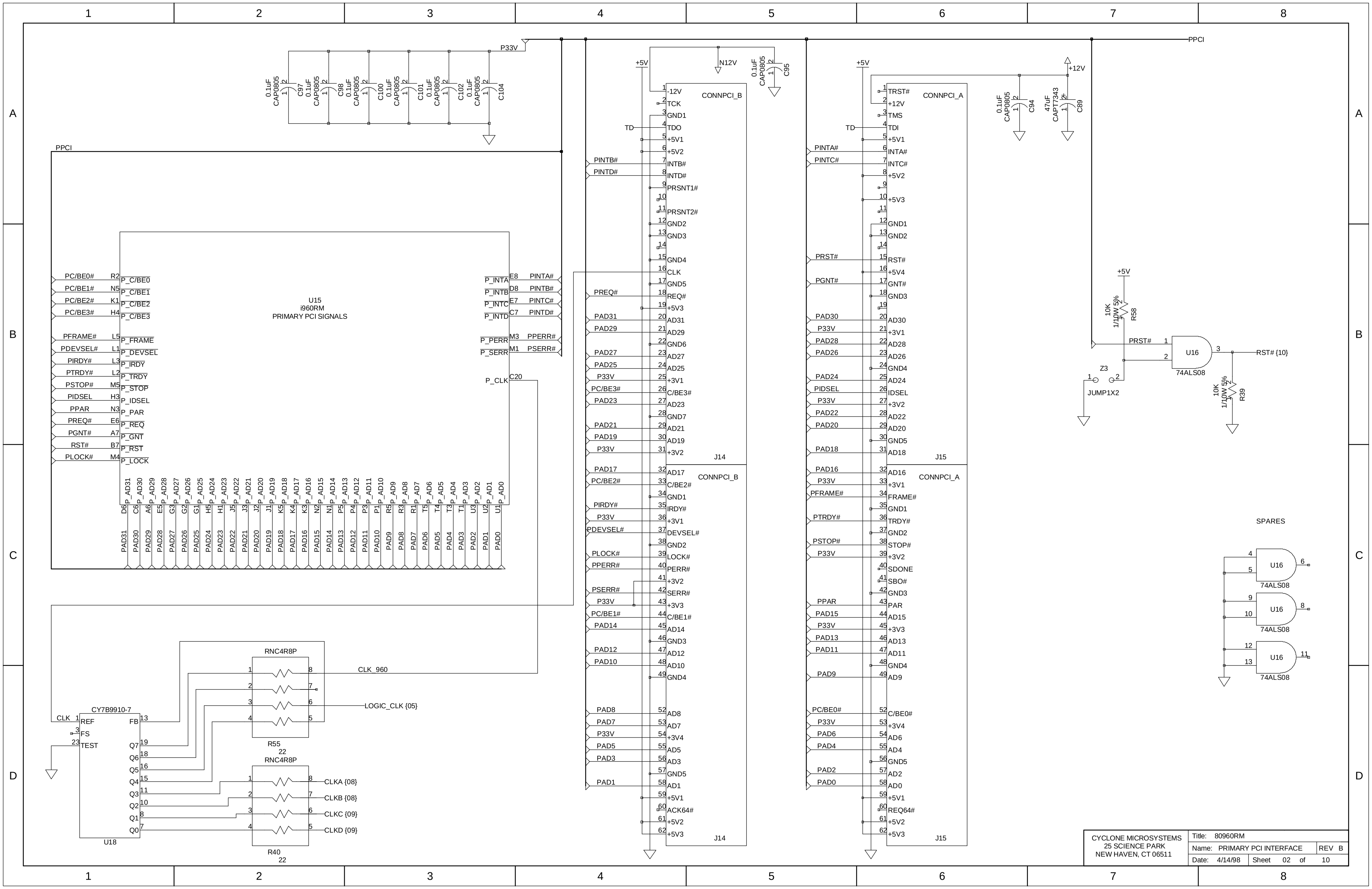
PART # 101-1950-01
U19
PALLV16V8Z-20JI
SCKE0 {03,06}
SCKE1 {03,06}
RAM3V {01,04,06}
RST# {02}
+12V
MAX712
U14
THI
V+
VLIMIT
REF
TEMP
CC
TLO
GND
DRV
BATT+
PGM0
PGM1
PGM2
PGM3
FASTCHG
BATT-
2N6109
Q4
CMR1-02
CR8
BATTERY
BT1
BT2
BT3
BT4
BT5
BT6
BT7
BT8
BATT_HLDR
MAX1651
U8
SHDN
FB
EXT
CS
OUT
SI9430
Q1
MBRS340T3
CR7
47uH
L1
SPARES
U10
LM339
+5V
FAN CONN
PWR
GND
FAN
J13
NOTE: VCC FOR LM339 IS +5V
IRQFAN# {07}
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RN
Name: BATTERY/MONITOR
REV B
Date: 4/14/98
Sheet 11 of 11

Table B-2. IQ80960RM Schematics List

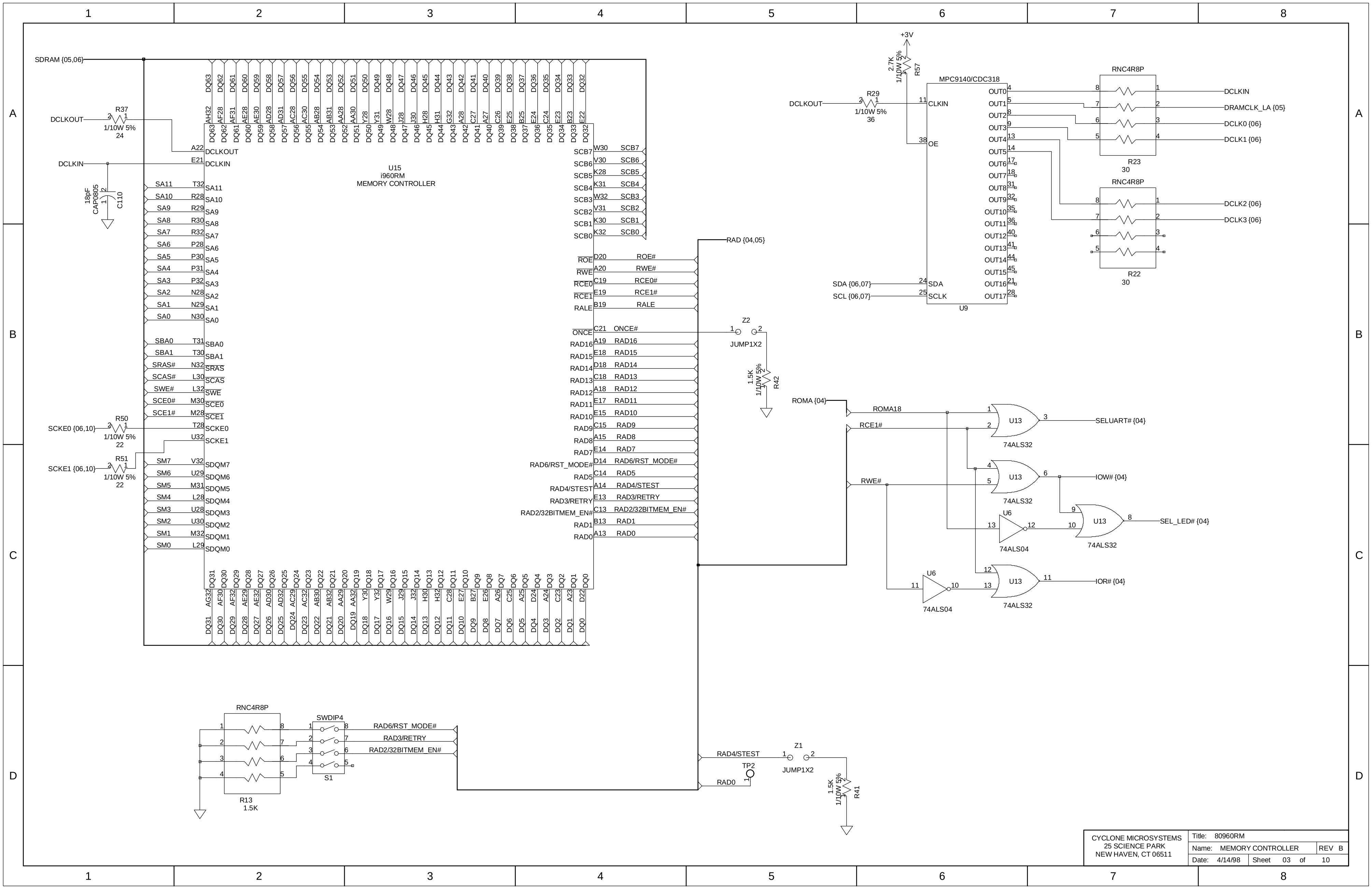
Page	Schematic Title
B-14	Decoupling and 3.3V Power
B-15	Primary PCI Interface
B-16	Memory Controller
B-17	Flash ROM, UART, & LEDs
B-18	Logic Analyzer I/F
B-19	SDRAM 168-Pin DIMM
B-20	Secondary PCI/960 Core
B-21	Secondary PCI Bus 1/2
B-22	Secondary PCI Bus 3/4
B-23	Battery/Monitor

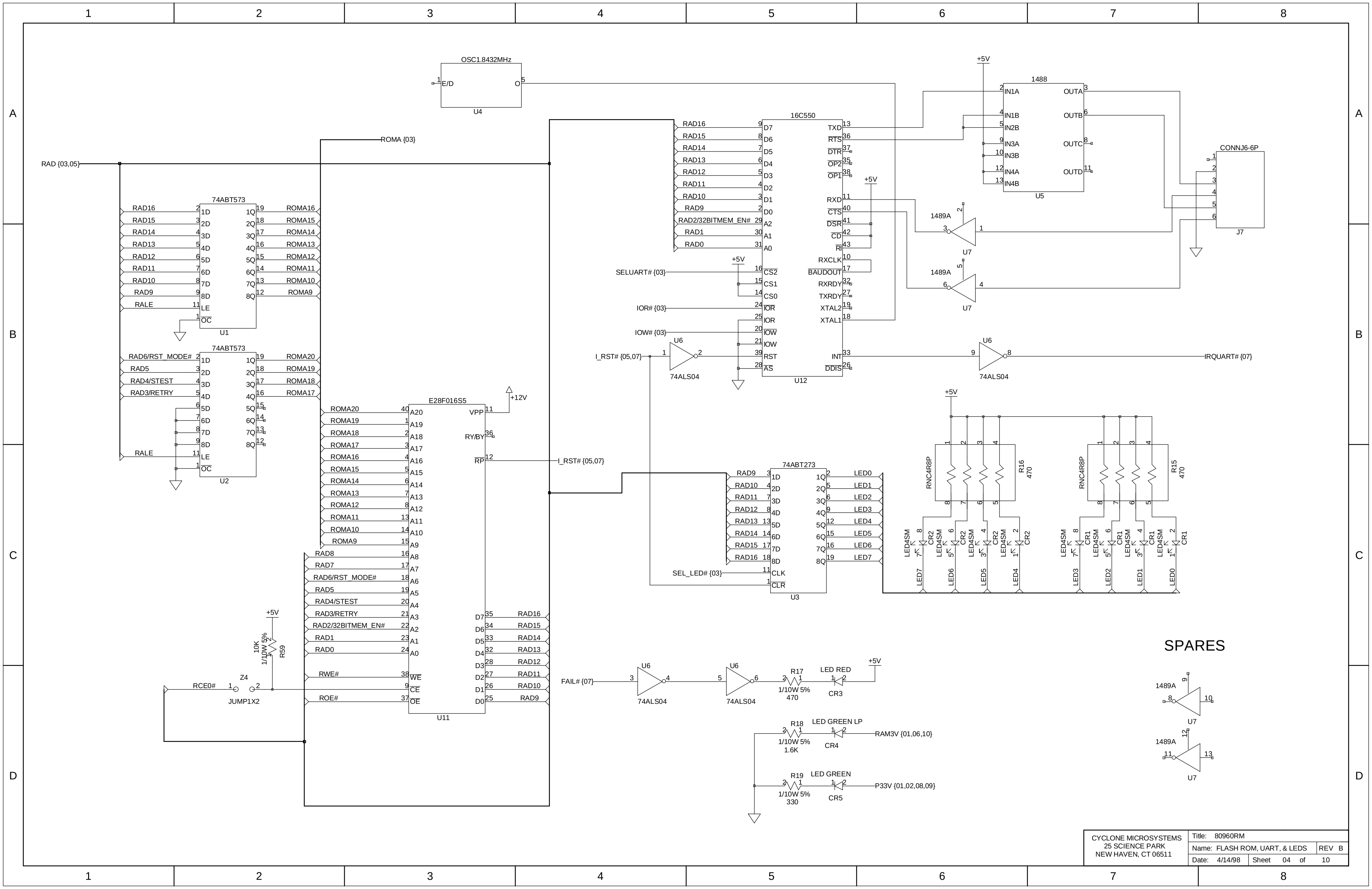


CYCLONE MICROSYSTEMS 25 SCIENCE PARK NEW HAVEN, CT 06511	Title: 80960RM	
	Name: DECOUPLING & 3.3V POWER	REV B
	Date: 4/14/98	Sheet 01 of 10

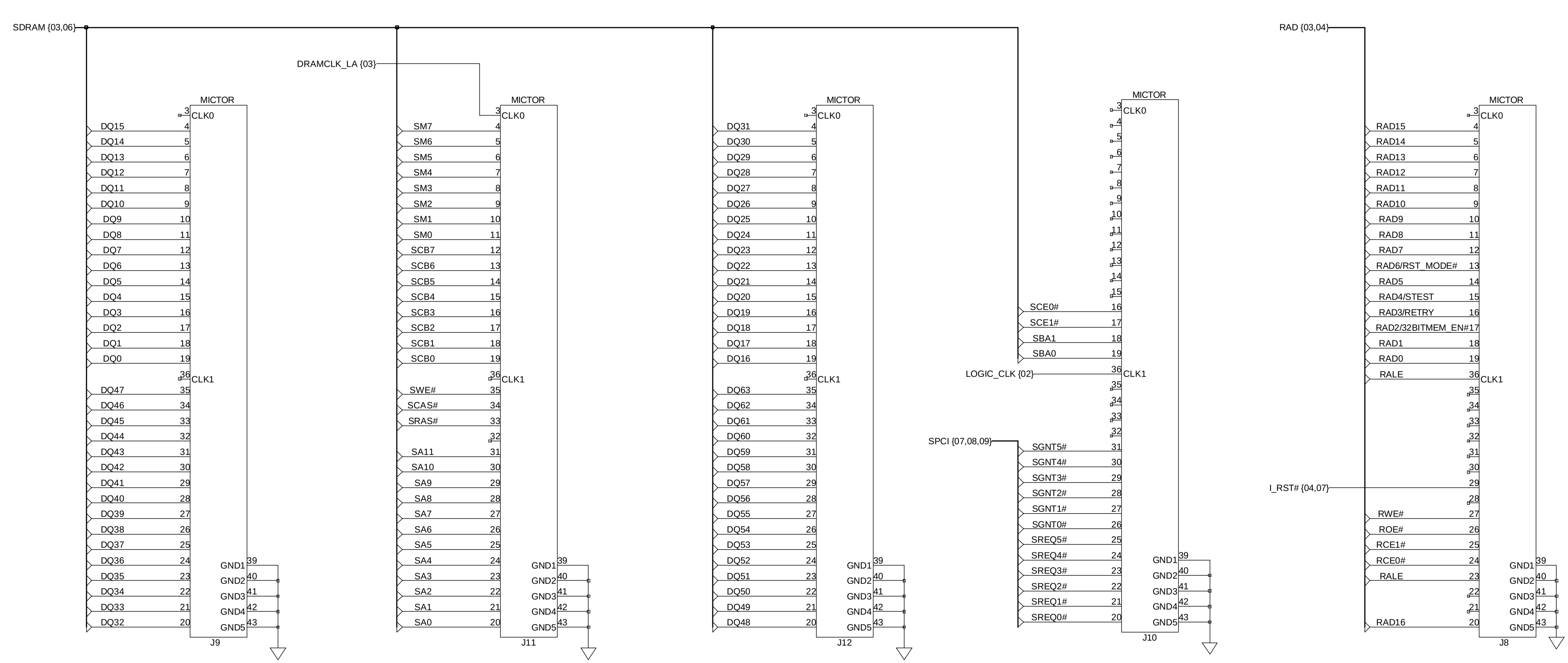


CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RM
Name: PRIMARY PCI INTERFACE
REV B
Date: 4/14/98
Sheet 02 of 10
U15
i960RM
PRIMARY PCI SIGNALS
CONNPCI_B
J14
CONNPCI_A
J15
CY7B9910-7
U18
RNC4R8P
R55
22
R40
CLK_960
LOGIC_CLK {05}
CLKA {08}
CLKB {08}
CLKC {09}
CLKD {09}
U16
74ALS08
RST# {10}
PRST#
R58
R39
10K
1/10W 5%
Z3
JUMP1X2
SPARES
PPCI
P33V
+5V
+12V
N12V
0.1uF
CAP0805
C94
C95
C97
C98
C100
C101
C102
C104
47uF
CAPT7343
C89

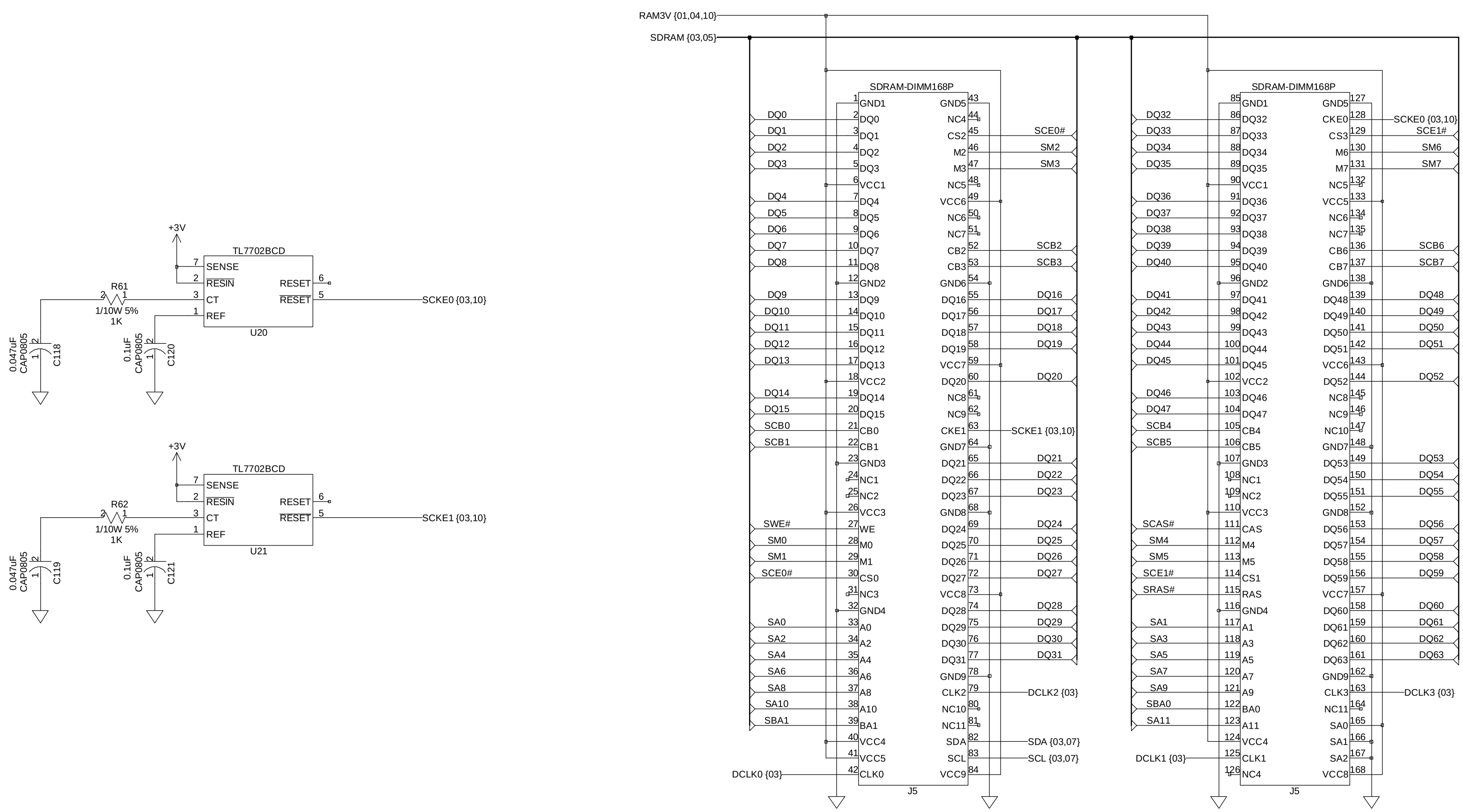

U15
i960RM
MEMORY CONTROLLER
MPC9140/CDC318
U9
RNC4R8P
R23
R22
R13
SWDIP4
S1
U13
74ALS32
U6
74ALS04
JUMP1X2
SDRAM {05,06}
DCLKOUT
DCLKIN
RAD {04,05}
ROMA {04}
SDA {06,07}
SCL {06,07}
SCKE0 {06,10}
SCKE1 {06,10}
DRAMCLK_LA {05}
DCLK0 {06}
DCLK1 {06}
DCLK2 {06}
DCLK3 {06}
SELUART# {04}
IOW# {04}
SEL_LED# {04}
IOR# {04}
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RM
Name: MEMORY CONTROLLER
REV B
Date: 4/14/98
Sheet 03 of 10



OSC1.8432MHz
U4
16C550
U12
1488
U5
CONNJ6-6P
J7
1489A
U7
U6
74ALS04
IRQUART# {07}
74ABT573
U1
U2
E28F016S5
U11
74ABT273
U3
RNC4R8P
R16
R15
470
LED4SM
CR1
CR2
ROMA {03}
RAD {03,05}
SELUART# {03}
IOR# {03}
IOW# {03}
I_RST# {05,07}
SEL_LED# {03}
+12V
+5V
RCE0#
Z4
JUMP1X2
R59
10K
1/10W 5%
FAIL# {07}
R17
LED RED
CR3
R18
1.6K
LED GREEN LP
CR4
RAM3V {01,06,10}
R19
330
LED GREEN
CR5
P33V {01,02,08,09}
SPARES
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RM
Name: FLASH ROM, UART, & LEDS
REV B
Date: 4/14/98
Sheet 04 of 10

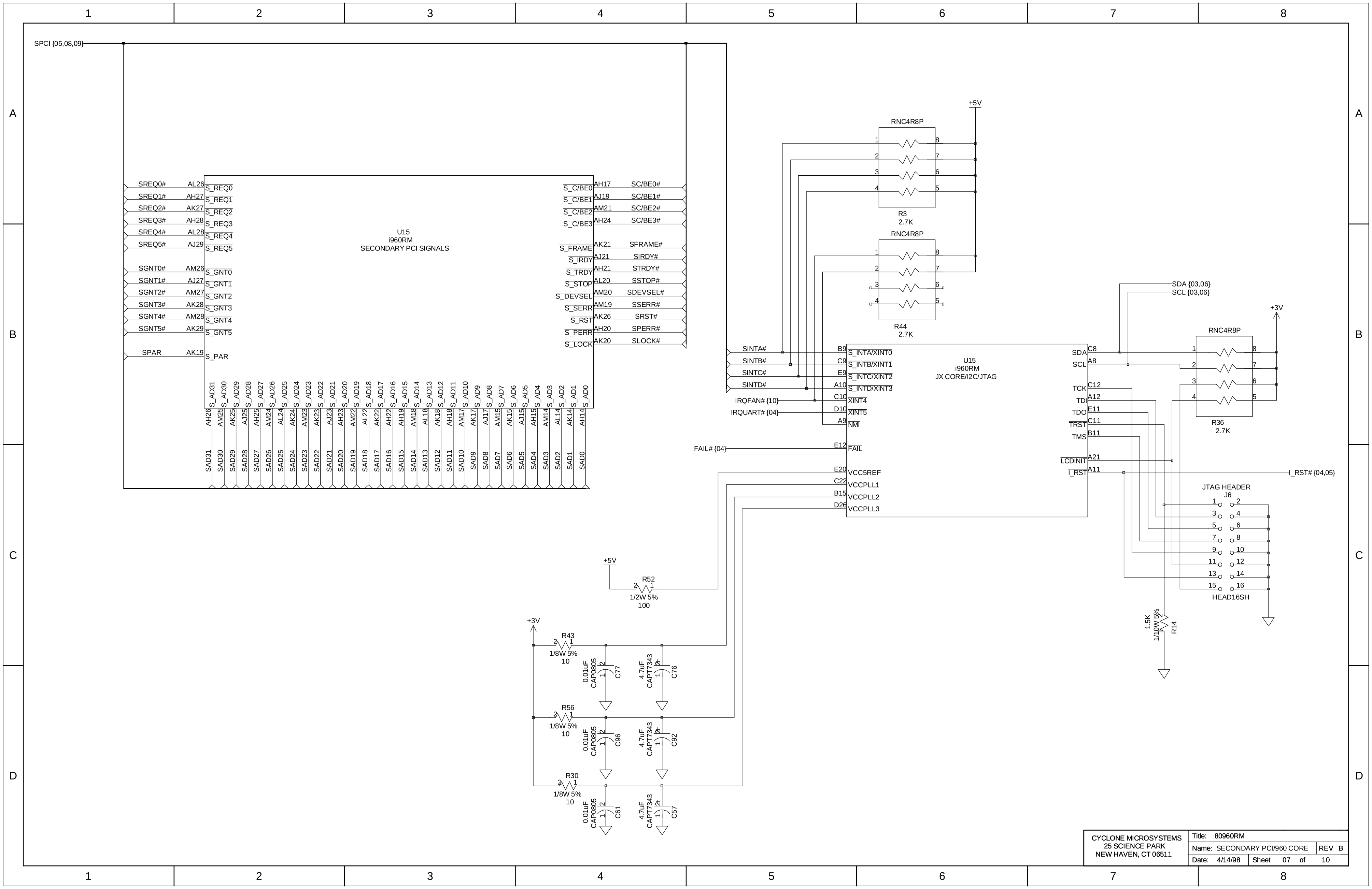


CYCLONE MICROSYSTEMS 25 SCIENCE PARK NEW HAVEN, CT 06511	Title: 80960RM		
	Name: LOGIC ANALYZER I/F		REV B
	Date: 4/14/98	Sheet 05 of 10	

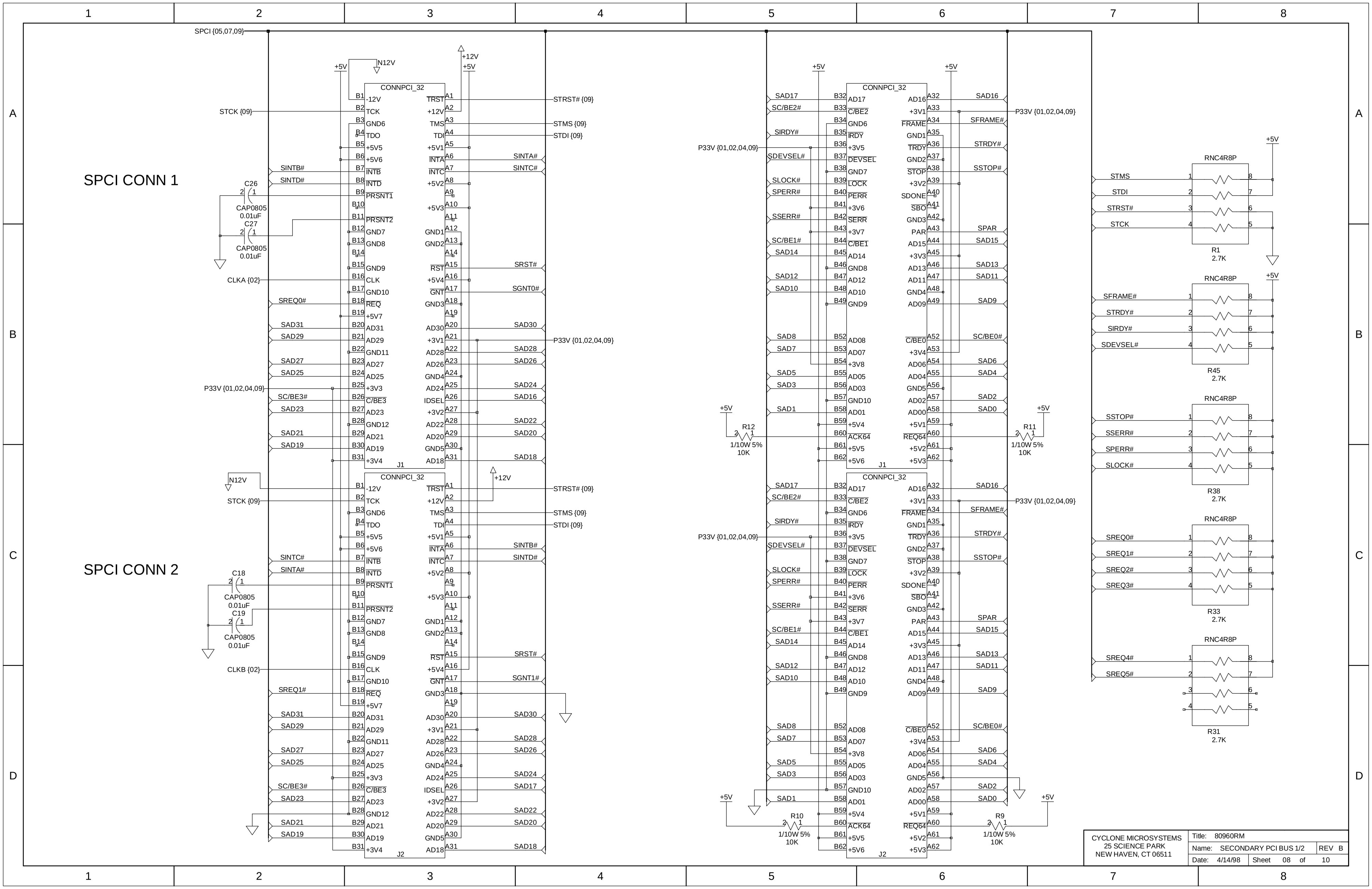


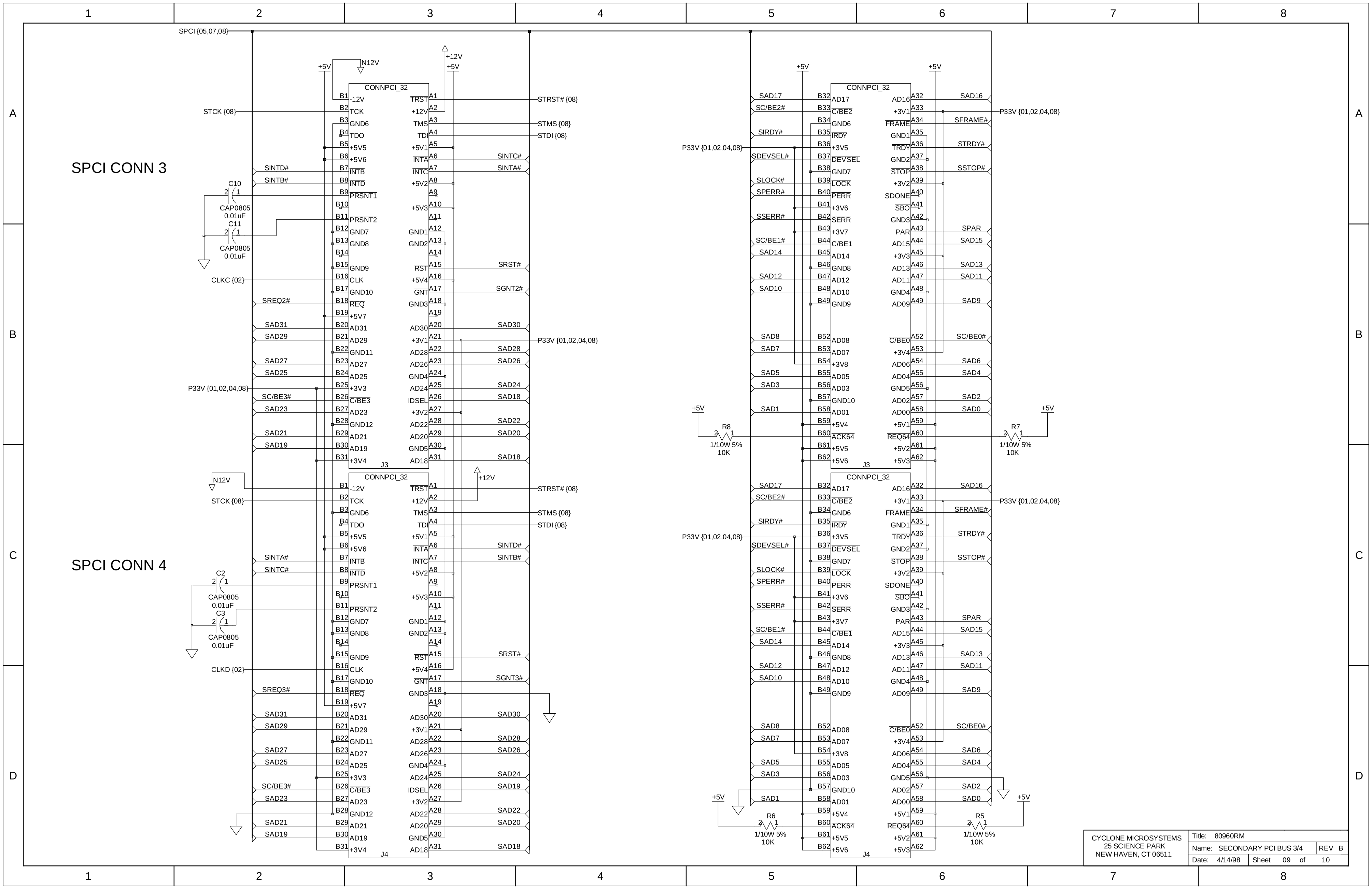
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511

Title: 80960RM			
Name: SDRAM 168-PIN DIMM			REV C
Date: 2/4/99	Sheet 6	of	10

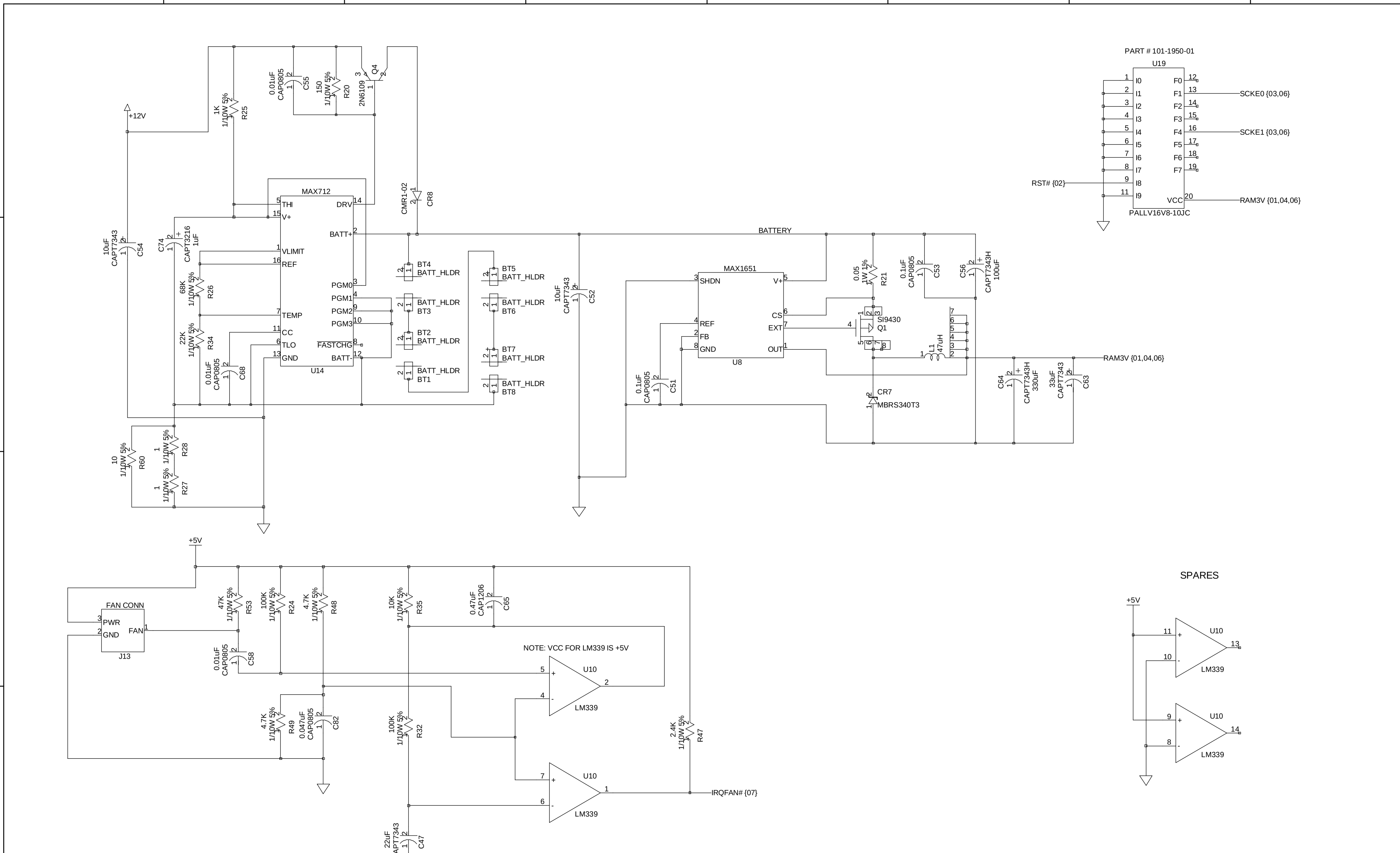


SPCI {05,08,09}
U15
i960RM
SECONDARY PCI SIGNALS
SREQ0# AL26 S_REQ0
SREQ1# AH27 S_REQ1
SREQ2# AK27 S_REQ2
SREQ3# AH28 S_REQ3
SREQ4# AL28 S_REQ4
SREQ5# AJ29 S_REQ5
SGNT0# AM26 S_GNT0
SGNT1# AJ27 S_GNT1
SGNT2# AM27 S_GNT2
SGNT3# AK28 S_GNT3
SGNT4# AM28 S_GNT4
SGNT5# AK29 S_GNT5
SPAR AK19 S_PAR
S_C/BE0 AH17 SC/BE0#
S_C/BE1 AJ19 SC/BE1#
S_C/BE2 AM21 SC/BE2#
S_C/BE3 AH24 SC/BE3#
S_FRAME AK21 SFRAME#
S_IRDY AJ21 SIRDY#
S_TRDY AH21 STRDY#
S_STOP AL20 SSTOP#
S_DEVSEL AM20 SDEVSEL#
S_SERR AM19 SSERR#
S_RST AK26 SRST#
S_PERR AH20 SPERR#
S_LOCK AK20 SLOCK#
S_AD31 AH26 SAD31
S_AD30 AM25 SAD30
S_AD29 AK25 SAD29
S_AD28 AJ25 SAD28
S_AD27 AH25 SAD27
S_AD26 AM24 SAD26
S_AD25 AL24 SAD25
S_AD24 AK24 SAD24
S_AD23 AM23 SAD23
S_AD22 AK23 SAD22
S_AD21 AJ23 SAD21
S_AD20 AH23 SAD20
S_AD19 AM22 SAD19
S_AD18 AL22 SAD18
S_AD17 AK22 SAD17
S_AD16 AH22 SAD16
S_AD15 AH19 SAD15
S_AD14 AM18 SAD14
S_AD13 AL18 SAD13
S_AD12 AK18 SAD12
S_AD11 AH18 SAD11
S_AD10 AM17 SAD10
S_AD9 AK17 SAD9
S_AD8 AJ17 SAD8
S_AD7 AM15 SAD7
S_AD6 AK15 SAD6
S_AD5 AJ15 SAD5
S_AD4 AH15 SAD4
S_AD3 AM14 SAD3
S_AD2 AL14 SAD2
S_AD1 AK14 SAD1
S_AD0 AH14 SAD0
+5V
RNC4R8P
R3
2.7K
RNC4R8P
R44
2.7K
U15
i960RM
JX CORE/I2C/JTAG
SINTA# B9 S_INTA/XINT0
SINTB# C9 S_INTB/XINT1
SINTC# E9 S_INTC/XINT2
SINTD# A10 S_INTD/XINT3
IRQFAN# {10} C10 XINT4
IRQUART# {04} D10 XINT5
A9 NMI
FAIL# {04} E12 FAIL
E20 VCC5REF
C22 VCCPLL1
B15 VCCPLL2
D26 VCCPLL3
SDA C8
SCL A8
TCK C12
TDI A12
TDO E11
TRST C11
TMS B11
LCDINIT A21
I_RST A11
I_RST# {04,05}
SDA {03,06}
SCL {03,06}
+3V
RNC4R8P
R36
2.7K
JTAG HEADER
J6
HEAD16SH
R14
1.5K
1/10W 5%
+5V
R52
1/2W 5%
100
+3V
R43
1/8W 5%
10
R56
1/8W 5%
10
R30
1/8W 5%
10
0.01uF CAP0805 C77
4.7uF CAPT7343 C76
0.01uF CAP0805 C96
4.7uF CAPT7343 C92
0.01uF CAP0805 C61
4.7uF CAPT7343 C57
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RM
Name: SECONDARY PCI/960 CORE
REV B
Date: 4/14/98
Sheet 07 of 10


SPCI CONN 1
SPCI CONN 2
J1 CONNPCI_32
J2 CONNPCI_32
R1 2.7K RNC4R8P
R45 2.7K RNC4R8P
R38 2.7K RNC4R8P
R33 2.7K RNC4R8P
R31 2.7K RNC4R8P
R9 1/10W 5% 10K
R10 1/10W 5% 10K
R11 1/10W 5% 10K
R12 1/10W 5% 10K
C26 CAP0805 0.01uF
C27 CAP0805 0.01uF
C18 CAP0805 0.01uF
C19 CAP0805 0.01uF
SPCI (05,07,09)
P33V (01,02,04,09)
CLKA (02)
CLKB (02)
STCK (09)
STMS (09)
STDI (09)
STRST# (09)
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RM
Name: SECONDARY PCI BUS 1/2
Date: 4/14/98
Sheet 08 of 10
REV B



SPCI CONN 3
SPCI CONN 4
CONNPCI_32
J3
J4
SPCI {05,07,08}
STCK {08}
STRST# {08}
STMS {08}
STDI {08}
CLKC {02}
CLKD {02}
P33V {01,02,04,08}
C10 CAP0805 0.01uF
C11 CAP0805 0.01uF
C2 CAP0805 0.01uF
C3 CAP0805 0.01uF
R5 1/10W 5% 10K
R6 1/10W 5% 10K
R7 1/10W 5% 10K
R8 1/10W 5% 10K
CYCLONE MICROSYSTEMS
25 SCIENCE PARK
NEW HAVEN, CT 06511
Title: 80960RM
Name: SECONDARY PCI BUS 3/4
REV B
Date: 4/14/98
Sheet 09 of 10



CYCLONE MICROSYSTEMS 25 SCIENCE PARK NEW HAVEN, CT 06511	Title: 80960RM	
	Name: BATTERY/MONITOR	REV B
	Date: 4/14/98	Sheet 10 of 10

PLD Code C

```
MODULE BATT

//TITLE         SDRAM Battery Backup Enable
//PATTERN       101-1809-01
//REVISION
//AUTHOR        J. Neumann
//COMPANY       Cyclone Microsystems Inc.
//DATE          10/30/97
//CHIP          PALLV16V8Z-20JI
// 1/20/98 Modify target device to PALLV16V8Z-20JI

//Initial release.

        PRSTn       PIN 9;//Primary PCI reset
        SCKE0           PIN 13;  //SDRAM bank 0 clock enable
        SCKE1           PIN 16; //SDRAM bank 1 clock enable
        OUT0            PIN 14;  //SCKE0 output enable
        OUT1            PIN 17; //SCKE1 output enable

EQUATIONS

// If SDRAM clock enable goes low, SDRAM clock enable
// must be held low to ensure that the SDRAM is held in auto refresh mode.
// Reset going high will release the hold on SCKE.

        OUT0 = SCKE0.PIN & PRSTn        //SCKE is the set term, PRSTn is the reset term
                # SCKE0.PIN & OUT0.PIN
                # !SCKE0.PIN & PRSTn;

        SCKE0 = 0;
        SCKE0.OE = !OUT0;               //When OUT = 0, SCKE is grounded
                                        //When OUT = 1, SCKE is high impedance

        OUT1 = SCKE1.PIN & PRSTn
                # SCKE1.PIN & OUT1.PIN
                # !SCKE1.PIN & PRSTn;

        SCKE1 = 0;
        SCKE1.OE = !OUT1;

END
```

Recycling the Battery D

The IQ80960RM/RN platform contains four AA NiCd batteries. Each battery has the logo of the Rechargeable Battery Recycling Corporation (RBRC) stamped on it. The recycling fees have been prepaid on these batteries. Do not dispose of a rechargeable battery with regular trash in a landfill. Rechargeable batteries contain toxic chemicals and metals that are harmful to the environment. Improperly disposing of rechargeable batteries is also illegal. The RBRC logo on a battery is a verification that recycling fees have been prepaid to the RBRC and such a battery can be recycled at no additional cost to the user. The RBRC is a non-profit corporation that promotes the recycling of rechargeable batteries, including NiCd batteries.

Information on the RBRC program and the locations of participating recycling centers can be obtained by telephoning 1-800-8-BATTERY (in the USA), and following the recorded instructions. The information obtained from this telephone number is updated frequently, since the RBRC program is growing, the new recycling locations are being added regularly.